Movicon® CE

COMPACT SCADA/HMI FOR WINDOWS CE

Visualization and control for embedded HMI based on Windows CE.

Movicon CE Reference Guide

Version 11.3 - Ed. Feb. 2012

Cod. DOCS 11CE-E Build 1101

Table Of Contents

1. PREFACE 3

1.1. INTRODUCTION 3
1.2. READ ME FIRST 4

2. GENERAL 7

2.1. GENERAL CONCEPTS 7
2.2. LICENCES 7
2.3. RESTRICTIONS 9
2.4. PROJECT CONSTRAINTS 13

3. HARDWARE TYPES 15

3.1. HARDWARE 15
3.2. SUPPORTED DEVICES 16

4. INSTALLATION 19

4.1. MS ACTIVESYNC AND WINDOWS MOBILE DEVICE CENTER 21
4.1.1. Serial Connections 22
4.1.2. Serial Cable for MS ActiveSync and Windows Mobile Device Center 23
4.1.3. Network Connections 23

5. EDITING 25

5.1. EDITING GENERAL INFORMATION 25
5.2. DESIGNING PROJECTS 26
5.3. CONFIGURATION FILE 28
5.4. PROJECT UPLOAD/DOWNLOAD 31
5.5. CREATING CUSTOM PLUGINS 37
5.6. RAM USE 41

6. MEMORY USAGE WITH HISTORICALS 45

6.1. MEMORY USE WITH IMDB 45
6.2. MEMORY USE WITH SQL SERVER MOBILE 47
6.3. RDA (REMOTE DATA ACCESS) 50

7. DESIGNING GUIDELINES 51

7.1. VARIABLES 51
7.2. THE GRAPHICS 51
7.3. ALARM MANAGEMENT 55
7.4. HISTORICAL LOG 56
7.5. DATA LOGGERS AND RECIPES 56
7.6. TRENDS 57
7.7. GRID 57
7.8. COMMUNICATION DRIVERS 58
7.9. OPC 59
7.10. LOGICS 59
7.11. NETWORKING 60
7.12. WEB CLIENT 60

8. RUNNING 61

8.1. RUNNING PROJECTS 61
8.2. PROJECTS DEBUG 63

1. Preface

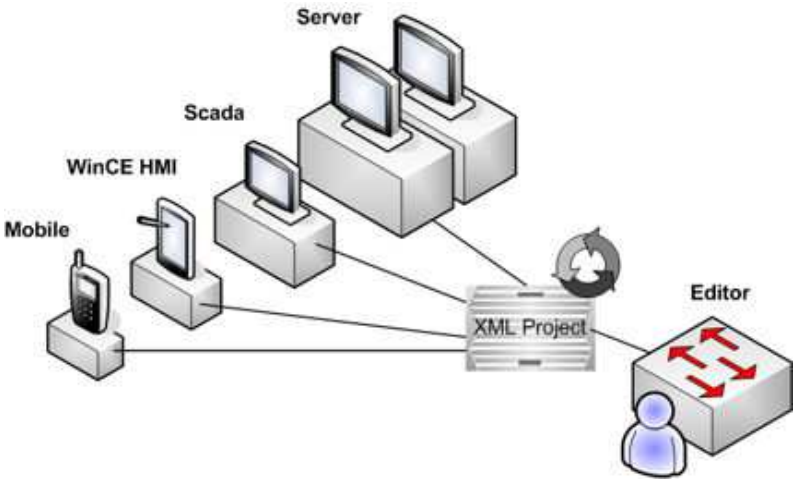
1.1. Introduction

Movicon CE is the Movicon RunTime version for embedded Microsoft Window CE embedded operating systems. Movicon allows projects to be exported to Windows CE platforms and run with the Movicon CE Runtime module.

Movicon CE is the new generation of compact Scada/HMI for Windows CE. Thanks to the new Movicon technology the compact version (Compact HMI) of the Movicon supervision and control software also adopts the XML technologies and Web Services to guarantee enhanced performances and potentiality in the world of Pocket systems and display terminals.
Movicon CE supports the Microsoft PocketPC (Windows Mobile) and Embedded PC, version 5.x and later (WinCE.Net), operating systems.

Thanks to Movicon one-only development environment you can get scalability when creating applications that can run either in Win32/64 or WinCE.

The Movicon CE runtime engine executes the same Movicon XML projects. Your projects can therefore be run on your HMI device, on display terminals, on PDA, Pocket PCs, mobile systems, Wireless... (Pocket PC and SSDK devices).
One-only project, the same file, run on different platforms, thanks to the project's XML structure.



The advantages you get from using Movicon CE are:

Integrated applied project openness with the same software on different hardware terminals.
The advantage of being able to keep the software while changing operating panels allowing you to choose the product most suitable for your needs. For instance, the same project can be run on panels like Proface or Advantech or Vipa, Asem, ESA, Suetron and others.

Integrated machine information openness with product line or higher system levels.
The advantage of being able to freely have information circulation thanks to the Ethernet network and OPC technologies. All WinCE terminals have a LAN port integrated.

Increase in graphic potentiality.
Movicon CE can be termed as a "small Scada" with big integrated potentiality. Powerful graphics, powerful alarm management, powerful historicals with relational DB management which can be integrated on the net, trends, recipes, scripts, sending SMS. All ready to be used in a powerful but simple programming environment such as Movicon.

Cost Reductions.
Movicon CE therefore permits the use of one-only software in your company business, whether for supervision on PCs or for display terminals, saving you money in training, personnel and maintenance. Thanks to Movicon CE's extreme openness you will be able to reduce costs in terminals by choosing those most suited to your needs while having the benefit of keeping and using the same software.

1.2. Read Me First

Before starting a project for devices based on Windows CE you should check that the used device's capacity and carefully read the warnings below, the product documentation and guidelines for designing projects.

IMPORTANT:

BEFORE DESIGNING PROJECTS FOR DEVICES BASED ON WINDOWS CE WE STRONGLY RECOMMEND THAT YOU CHECK YOUR HARDWARE CAPACITY (MEMORY AVAILABLE AND CPU TYPE) AND CHECK ANY RESTRICTIONS AS INDICATED IN THE DOCUMENTATION ON "DESIGNING GUIDELINES FOR WINCE TOUCH SCREENS".

IMPORTANT:

YOU SHOULD ALWAYS KEEP IN MIND THAT DEVICES BASED ON WINDOWS CE HAVE MORE RESTRICTIONS THAN SYSTEMS BASED ON Win32/64 BOTH AS HARDWARE AND AS OPERATING SYSTEMS.
THEREFORE IT IS IMPORTANT THAT YOU DESIGN YOUR PROJECT IN PARITY WITH THE CAPACITY OF THE SYSTEM YOU ARE USING.

THE MOVICON RESTRICTIONS FOR WinCE COMPARED TO THE NORMAL VERSION FOR Win32/64 HAVE BEEN HIGHLIGHTED IN THIS PRODUCT DOCUMENTATION, LEAVING THE DESIGNER TO CHECK THE RESTRICTIONS OF THE HARDWARE AND OPERATING SYSTEM THEY ARE USING.

MOVICON EDITOR IS THE SAME ONE USED FOR Win32/64 and WinCE PROJECTS. EVEN THOUGH THIS IS A GREAT ADVANTAGE FOR THE DESIGNER IT MIGHT BE RISKY WHEN USING THE SAME POTENTIALITIES INDEPENDENTLY FROM THE TARGET DEVICE USED.
BEFORE PROJECT DESIGNING TAKE NOTE OF THE DIFFERENCES IN RESTRICTIONS OF SYSTEM BASED ON WINDOWS CE AND THOSE OF THE VARIOUS DEVICES AVAILABLE ON THE MARKET.

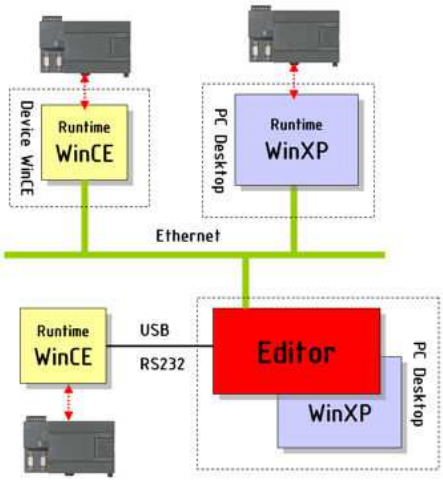
WARNING:
Before designing projects for Windows CE, it would be in your favour to create a 'new' project by Movicon choosing the Windows CE as destination platform. Even though this is not obligatory as the project created will however be executable in Windows CE, it will however autoconfigure the new project by resetting some of the general properties correctly. In addition to this, a .ini file will be created in the project folder containing the default constraints which will be assigned to the project in editing mode (see relating chapter).

You can select the "WinCE" as the project's supported platform afterwards from the project's property options. Also remember that Movicon also multiple selections, where the most powerful platform selected will be considered.

2. General

2.1. General Concepts

Movicon CE is a runtime engine for Microsoft Windows CE embedded operating systems. The Movicon version supports Windows CE operating systems in versions 5.0 or later.
The user can use the Movicon development environment for creating supervision applications which can be interpreted and executed on Movicon CE runtime engines. The Movicon editor permits you to create projects built with XML files which can be run on both Desktop PCs and CE devices.



The programmer can then work using the Desktop PC with the same Movicon platform normally used. However, you must keep in mind that the WinCE device is not capable of completely supporting all the potentialities offered by Movicon and typical of supervision systems for Desktop PCs. Therefore you need to take into account the restrictions deriving from the operating system's scarce capacity and from the device hardware when creating projects for CE devices.

To ensure that the programmer is able to design only what the Windows CE is capable of supporting, the destination platform type should be selected from the Project's properties making sure that only WinCE is set up as platform.

2.2. Licences

Devices using Windows CE provide a licence management in the form of a **Softkey** Unlock code only.
There are no hardware keys of any type available. The project runtime will run in Demo mode on the WinCE device only without a license.

Movicon CE license management

When purchasing a Movicon CE you will receive it in the form of a Serial Number which gives you **access rights** to the Progea website to get the Softkey unlock code to be inserted into your device. In this way, you can access the website at any time, 24 hours a day, to automatically create the unlock code according to your website code.

Follow the step-by-step procedure in order to get and insert your softkey:

1. When purchasing the license (or purchasing a CE device already integrated with a license to be activated), you will receive a serial number and access rights to the Progea management website (www.progea.com or http://support.progea.com/softkey/)
2. Startup Movicon on the CE device. Movicon will display a window to be used for entering the license or for running Movicon CE in demo mode
3. Go to and access the Progea Website using your access code, then follow the instructions to insert the Site Code with which the license (SoftKey) will automatically be created
4. Jot down the Softkey license and insert it into the appropriate edit box which appears at the Movicon Startup
5. Many devices require you to save the WinCE Registry so that data is kept memorized permanently. In any case, the license is linked to the device and will not change upon reinstallation

- **The instructions you received along with your purchase or those from the supplier should be respected.**

The Movicon CE license exists in two versions (Standard or Lite).

The Lite version has some technical restrictions:

- Max. 256 Alarms
- Max. 1 Web Client Users
- Max. 1 communication driver
- Max. 1024 bytes of variables in use
- Max. 8 DataLoggers
- IL and VBA Logics (max. 128 kb)
- Networking: NO (project can be only Network Server)

The Full version has the following technical restrictions:

- Max. 1024 Alarms
- Max. 2 Web Client Users
- Max. 4 communication driver (not more than 2 recommendable)
- Max. 4096 bytes of variables in use
- Max. 8 DataLoggers
- IL and VBA Logics (max. 128 kb)
- Networking, Max 256 connections (project can be Network Server or Client)

IMPORTANT:
APART FROM THE LICENSE TYPE USED AND ITS RESTRICTIONS, THE DESIGNER SHOULD ALSO CHECK THE RESTRICTIONS OF THE HARDWARE DEVICE BEING USED: THE FUNCTION RESTRICTIONS OF THE MOVICON PRODUCT LICENSE ARE GENERIC, BUT THE HARDWARE MAY HAVE ITS OWN DESIGNING RESTRICTIONS INDEPENDENTLY FROM THE LICENSE TYPE USED.

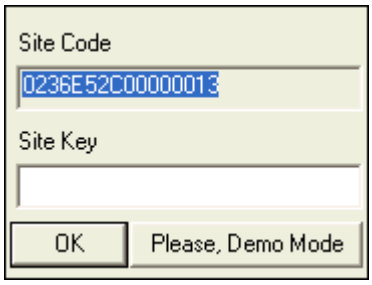
Movicon CE Site Code Generation

The site code in Movicon CE is generated in one of the following ways:

1. Using the network card's mac address. Movicon CE uses a WinCE API to get the list of network cards installed in the device, and uses the mac address of the first network card returned by this function.
2. If the device does not have any network card installed, the site code will then be generated using the panel's "Device ID.
3. If the "Device ID" cannot be found, the site code will be generated at random.

Inserting the Movicon CE license

If an invalid or no license has been inserted at startup, Movicon CE will display a window for selecting the start mode.
When no license has been preinstalled by the device builders, you can get the license based on the "Site Code" displayed.
By using this Site Code (and the purchase serial number), you can get your runtime license according to the instructions received when purchasing the site key.
Your license, or "Site Key" must be inserted in the field underneath the Site Code, as shown below:



If you don't have a Site Key number to insert, you can still execute the project in Runtime mode by clicking the "Demo Mode" button in the same window. The project will be run for a limited two hours.

After having inserted the license (Site Key), remember to execute the command for saving the WinCE Registries (RegSave or its equivalent, depending on the device used).

Once the license has been inserted and saved in the WinCE System Registry, the window used for inserting the license will now show again when starting up Movicon CE. You can, however, make this window show by executing Movicon CE through the WinCE command line and specifying the "/i" o "-i" option. From the WinCE "Start - Run" menu execute:

MovCE.exe -i

This functionality may be needed in situations where the device's "Site - Code" needs to be reviewed or to execute a license update with a new license containing different characteristics.

Run in Demo Mode on Window CE

At project startup, Movicon CE will request you to enter the licence or press the Demo Mode button. When pressing the Demo Mode button the project will run in Demo mode. The project run is limited to 120 minutes in this mode. Messages are traced in the project log, at regular time intervals, indicating the remaining time left. The available options are:

- Screens = unlimited
- Alarms and Messages = unlimited
- I/O Bytes = 0
- Driver = 1
- Web Client users = 1
- All Options

2.3. Restrictions

Some functions that are not supported by the CE device should be kept under consideration when designing applications for WinCE devices. Although the non supported functions are appropriately "masked" when selecting WinCE as the target OS, it should however be remembered that a hardware device with limited calculation capacity and memory is being used and therefore the designer must always evaluate whether it is compatible with the objectives they wish to achieve.

Graphics Restrictions

Function	WinCE	Win32/64
Coloring	✔	✔
Linear Fillings	✔	✔
Polygon Fillings		✔

Rotations		✓
Dynamic X, Y Movements	✓	✓
Graphic Objects	✓	✓
Symbols Library	✓	✓
Button/Gauge with 3D Look		✓
Trend	✓	✓
Charts		✓
DB Grid	✓	✓
Embedded Screens	✓	✓

Other general graphics restrictions

- Composed drawing background (not solid)
- Dotted lines (solid and dashed) NB. when the line is thicker than 1 in respect to the drawing displayed on the desktop, where the line stays entirely inside, it will go over the border in the CE and will not show (no InsideFrame)
- Font escapement is not supported
- Drawing state is not supported. The Drawing State gives you the option to dull controls into light grey when the variable used by them does not have good quality. This feature is also available in Windows CE but instead of being dulled to light grey a small red square appears in the objects.
- EditPenProperties are supported but reduced
- Symbol Dragging is not support (because hardly used)
- The trend external setting files are not compatible between the desktop and the target PC
- The external Metafile drawings (WMF, EMF) are not supported in WinCE
- The PrePaint and PostPaint events pass a HDC without transforming the coordinates due to fact that Viewport is not supported
- Opening Screens as frames ("Open Frame (Multi-monitor)" command) is not fully supported in WinCE. The screen will open but will not remain in the foreground. When clicked in the area outside, the screen opened as frame will go into the background and become no longer visible.
- The scroll bars are not supported in Windows CE. Therefore screen scroll bars will not be displayed in projects run in Windows CE.
- ".ico" icon type images are not support in WinCE
- Images embedded in drawings are not supported

Alarms Restrictions

There are only restrictions in the notification and statistical analysis management.
Due to the fact that the Report Engine is not available in Windows CE, the commands for creating statistical reports on alarms are not available in the CE device.
The alarm notification functions are managed from the Alarm Dispatcher. Only the alarm notifications with SMS and E-mail are available in Windows CE as reported in the table below:

Function	WinCE	Win32/64
SMS - Via Modem GSM	✓	✓
SMS - Via SMMP		✓
Voice		✓
Fax		✓
E-mail - via TAPI		✓
E-mail - via SMTP	✓	✓
Alarm Statistics		✓

For further information please also refer to the paragraph on **"Alarm Management"**.

DataLogger/Recipe/Report Restrictions

The WinCE projects normally record the log files by using the IMDB technology (InMemory DB) which, different to ODBC/ADOCE, is much lighter and higher performing due to the XML text format used.
In situations where the ODBC has been selected in development mode (desktop) for recording log files, this will correspond to the ADOCE on the target as Windows CE does not support ODBC. Movicon will automatically convert the ODBC connections to ADOCE connections. In this way the projects on the target will have an open and standard historical log database.
The ADOCE data format is basted on Ms SQL Server CE.
Thanks to this automatic management, the project keeps the same identical characteristics in handling the historicals, but the database format on the target will be SQL Server CE type.

Always make sure that the CE device's memory capacity is sufficient enough to contain historical data in compliance to the amount setup in the project.

The Crystal Report Engine is not supported in Windows CE.

Function	WinCE	Win32/64
Data Loggers	✔	✔
Report Designer Engine		✔
Textual Report	✔	✔
Recipes	✔	✔
Crystal Report Engine		✔

WinWrap Basic Restrictions

Movicon CE guarantees support to the VBA language, therefore the code executed on the desktop is the same executable file on the target.

Nevertheless, it must be said that Progea cannot guarantee the complete support to all the countless functions, properties, methods and events available on the Desktop. It has not been possible, up till now, to efficiently test out every type of device with the correct execution of each single instruction.
The User is therefore advised to use the VBA code wisely by also always considering the device's calculation capacity on which the project is run.

The following listed functions are not supported on CE devices:

Function	WinCE	Win32/64
CurDir		✔

The Alarm Dispatcher is not supported by the Basic Script code management and therefore the SMS, E-Mail, etc functions cannot be used in Basic Script code.

IL Logic Restrictions

The IL Logic is perfectly compatible both with WinXP and WinCE. The deterministic features of WinCE are capable of managing logics inside the Movicon project with deterministic runtimes. Unfortunately, Progea has not carried out any specific tests on this and therefore cannot guarantee anything definite.

The "ILSleep" default value for WinCE is "50" milliseconds. But stays the same for Win32/64 at 10 milliseconds. A different value will allow free more CPU resources in projects using IL logic, especially in devices that have slow CPU. The default reduction of this parameter may be beneficial to user interface smoothness but IL execution may be a bit slower than before. This parameter can be changed when necessary.

Debugger Restrictions

The CE device's **Remote Debug** has been introduced from this version. When using the 'Attach to Process' function as described in the "Remote Project Debugging" section in the programmer's manual, you can "link" up to the process being run on the Target device from the Desktop, where you will be able to use all the Debugger On Line functions as well as for the remote device too.

Redundancy Restrictions

The Project redundancy feature is not supported in Movicon CE. This is mainly due to the restrictions suffered by the Windows operating system compact version.

Driver Restrictions

Communication drivers used in projects must be downloaded to device because this is not done by the Movicon CE installation.
he communication drivers used in the project are automatically installed on the device. It is absolutely necessary to verify whether the driver used in the project being edited is available in WinCE version.
The communication drivers for WinCE are purposely compiled for such operating system, however you need to check their availability. It is not always true that a driver for XP is also available in versions for WinCE. For instance, sometimes drivers use libraries of the PLC's constructors which might not be available in WinCE. Therefore always check driver availability before starting the project, by going through the table of drivers which is published and updated on the Progea Web Site mainly because they are continually being amplified, a driver which is initially unavailable may have been released or in the phase of being also released for WinCE. If in doubt, please contact the Progea offices or your dealer.

The driver for WinCE restrictions are:

Function	WinCE	Win32/64
Modem Telecontrol (TAPI)	To be verified	✔
VBA Interface	To be verified	✔
Multidriver	max. 2	✔

OPC Restrictions

Movicon CE supports the OPC technology as Client not as Server. Moreover the technology OPC Client XML is not supported.

Function	WinCE	Win32/64
OPC Client DA	✔	✔
OPC Server DA		✔
OPC Client XML DA		✔
OPC Server XML DA		✔

Web Server Restrictions

Movicon CE supports the Web Client technology. However, only two clients at a time are allowed contemporary access. This is to avoid the risk of having too many clients removing the already scarce quantity of resources, provided on the devices, all at once.
In addition to this you need to consider that a native Web Server is available in the devices as is with the Win32/64 which has a IIS (Internet Information Server) integrated. In this case too, access to the WinCE server for one Web Client user requires you to have the device specifications, supplied by its makers, at hand.
Access from a Web Client to a server based on Movicon CE can therefore be done in two ways:

1. Verify the whether there is a Web Server available on the device (if not, you can install one), configure it appropriately. In this case, access from Web Clients can be done by using the HTTP protocol and therefore through a normal Internet Browser
2. When there is not a web server, you can access as Web Client without using a browser, but by using the Web Client (JAVA) applet directly. In this case you will need to have the Web Client applet and a Java Virtual Machine installed

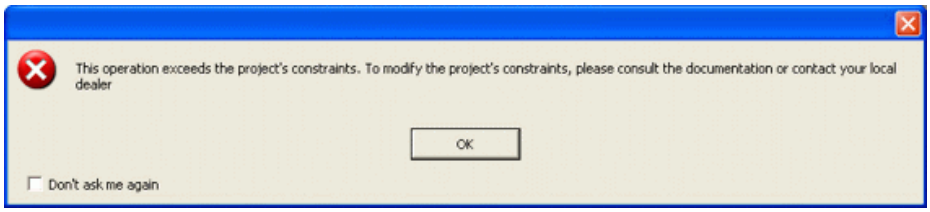
You can get further instructions on this by consulting the Movicon Web Client's User Manual.

Web Client Restrictions

A Web Client cannot be run from the Windows CE system by using a Web Browser or using the Java Machine directly with a command line due to the fact that a freely installable Java Machine, that is stable enough, does not currently exist for CE.

2.4. Project Constraints

In order to let the designer know about device constraints when creating projects for Windows CE, Movicon automatically creates a file in the project folder with the default designing constraints. These constraints are purely indicative and can be modified by clicking on the text file as shown below. It is then the designer who must valuate whether these constraints established for default can be modified in function with the hardware being used.
If these limits are exceeded the following window will automatically appear:


This operation exceeds the project's constraints. To modify the project's constraints, please consult the documentation or contact your local dealer
OK
Don't ask me again

For further information on this subject please refer to the "Project Constraints" section in the Movicon general Help

Programming contraints

When creating a project for Windows CE you will need to take into account the operating system's limitations as well as those mentioned below while creating the project:

Contraints for devices with less than 128 MBytes of memory or those using Windows CE 5.0.

Memory use in devices with Windows CE is usually very limited. We strongly advice you to read the following tips when using devices with Windows CE 5.0 or with Windows CE 6.0 which, however, has less than 128 MBytes of memory:

- **We advice you against using more than 2 objects with script code per screen and not more than 2 basic script resources in separate threads.** It would be best to not use or limit use of basic script code in designs. The WinWrap engine does in fact require 2.5 MBytes of memory + 300 KBytes for each object or resource containing basic script code + 700 KBytes if the basic script resource is checked in a separate thread. This memory cannot be allocated in extra process memory and therefore the process memory may exceed the 32 Mb limit set by the Windows CE 5.0. operating system. Eventually, if you cannot do without using script code, you should try to use it in one point only which could be the screen containing the objects.
- **Limit basic script code in alarm thresholds to not more than 10 alarms.**
- **Avoid using basic script code for managing communications.**
- **Do not use the IMDB in cases needing to record large amounts of data exceeding total of 5000 records the overall total amount allowed for the historicals: historical log, data logger, recipes, trace db.**
  We suggest you set the "Shared Table" option for each IMDB database as to not occupy process memory (limited to 32 MB in Windows CE 5.0).
- **Use images in bitmap format with max. resolution of 256 colours.** Movicon CE uses a cache for bitmap images used in drawings which once loaded will remain in memory even when screen is not loaded. Only those images set as screen background will get unloaded from memory when not used.

- **Disable the "Network Server" option in the Networking settings if the project has nothing to do as Server.**

Contraints for devices using Windows CE 5.0 or Windows CE 6.0.

In general, you need to take into account the follwong contraints for all devices using Windows CE:

- **You should limit the use of animations that need updating graphically every second and do not use more than 5 objects per screen (quick blink, moviments, resizing, animations with bitmaps, etc.).**
- **Make more use of objects from the Movicon toolbox (Object window) and less use of Templates from the Symbol Library which are mostly complex and contain basic script.** The reactivity of Template commands are in fact not as fast as those in standard objects.
- **Try to centralize the execution of IL Logic within your projects instead of having code portions spread out all over the place in different screens and symbols.**
- **Try avoid using Synapse logic.**
- **Avoid using the Windows CE panel as Server for more than 2 Web clients. Furthermore, the Web client page sizes should not exceed the panel's resolution.**
- **Avoid using the Windows Ce panel as Network Server for more than Network Clients.**
- **Avoid recording data at a frequency inferior to one record a second.**
- **Always check that there is sufficient remaining compact flash space for recording the total amount of historical data programmed in the project.**
- **Use the heap manager only when necessary, if the "_SysVar_:MemoryAvailVirtual" system variable nears or goes below the value of 5.242.880 bytes.** When going below this value, Movicon will enter into 'low memory' condition. Please refer to relating documentation.

3. Hardware types

3.1. Hardware

Please be reminded that Movicon is a powerful Scada/HMI platform designed for providing project designers just one unique and flexible designing tool for both powerful supervision systems based on Personal Computer (ie. Windows 32/64 bit) and small Touch Screen devices based on Windows CE at the same time.
The later requires the designer to carefully consider the target device's limitations, for which the project should be developed keeping the constraints in function with the panel's capacity in mind.
This document requires the designer to have the basic know-how of Movicon.

Touch Panels based on Windows CE usually vary in features. Therefore the designer should check whether their target device has at least the following requirements:

- Windows CE 5.0 or later
- A minimum CPU of 400 MHz
- A minimum RAM of 32 MB, 64 MB recommended

It is very important that the chosen hardware is suitable for the project you wish to design. Often mistakes are made in choosing Touch Panels for their price rather than their hardware features, which may turn out to be insufficient for project use.

- **The project must always be proportioned to the hardware used. Below are described how to choose the right device or how to adapt your project to the hardware available.**
- **Even though the Movicon software comes supplied preinstalled by the device builders, it does not mean that Movicon is a generic software and not designed expressly for one specific target. Therefore the tips mentioned in this documentation should always be taken seriously.**

Different WinCE hardware devices supported by Movicon

CPU type and graphics cards

The processor's calculation capacity is very important, and a good processor builds the foundations of obtaining good performances. It is important that you get good graphics cards for your panel's graphics. If processed program image outputs are sent to graphics cards not in parity with the processor being used, you may find a noticeable performance drop when refreshing displayed data or in the time it takes to change pages.

Benchmarks Performances

The below table shows the results relating to the number of graphic refreshes a second using a Movicon test project for displaying on screen the value of variables exchanged with the PLC (worst condition).

Nr. of graphic refresh a second using Movicon displays	90 Displays	60 Displays	30 Displays
VIPA TP606C (PXA 277)	0.9	1.88	4
ASEM OT1000 (Transmits i486)	4.29	6	20
Other devices...			

Chipcard Benchmark tool Graphic	Test Results
VIPA TP606C (PXA 277)	136 upd-sec
ASEM OT1000 (Transmits i486)	168 upd-sec
Suetron TP12C (PXA 255)	60 upd-sec
Other devices...	

Considerations

The following performances are normally considered acceptable:

HMI Panel expectancy Times
Page Change 0,5 - 1 sec.
Displayed Data Refresh 0,5 – 1 sec.
Command execution 0,5 – 1 sec.

HMI Panel expectancy	Max. Times acceptable
Page Change	0.5 - 1 sec.
Displayed Data Refresh	0.5 – 1 sec.
Command Execution	0.5 – 1 sec.

Movicon will allow the above performance indications to be obtained with ease providing you design your project to match the hardware type it is to run on. Obviously, this shouldn't be a problem in the designing phase when using powerful panels. However if the touch panel does not have the required hardware configurations you will find yourself having to solve problems to satisfy these constraints in order to adapt and optimize your project to the hardware resources provided in its designing phase. Movicon is a general-purpose software and therefore not tied just to one specific hardware and cannot fully ensure performances under the above mentioned circumstances.

Performances therefore depend on both project type and hardware type used which may result in being better or worst than those indicated on the table. This is not dependent on Movicon but on the system project designer.

3.2. Supported Devices

Movicon CE has been created so that is can be installed on more wider range of devices and/or operating terminals found on today's market. The automatic installation verifies the device's processor type and installs the corresponding runtime automatically.
Many processor types are supported (x86, VIA, Eden, Arm, StrogArm, MIPS, etc.), independently of the panel builders.
We suggest that, in any case, you verify the compatibility of the device, you intend, to use with the Movicon product. Full compatibility is guaranteed on:

1. Processor type
2. Platform type
3. Operating system image type

The below list of panel makers are those which have been tested directly by Progea. There are many more devices on the market which function just as well, and whose compatibility has been verified directly by the their makers or clients.

- ADS-Tech
- Advantech TPC
- Asem
- Beckhoff CX1000
- B&R PowerPanel
- Divus
- Exor
- ESA Elettronica
- Future Pad Fujitsu
- IPS
- Keba
- Mettler & Fuchs
- Pilz HMI
- Phoenix Contact
- ProFace
- ROI Computer
- System Copilot
- Sitek
- Sutron
- Techmark
- VIPA

PocketPC or palmtops:

- Compaq/HP (iPaq)
- Toshiba PDA
- Asus MyPAL
- Dell Axim X3

This list is updated on the Progea web site regularly (www.progea.com). For any further information on the type of devices used or the necessary tests, please contact the Progea offices or your local dealer.

4. Installation

When "MS ActiveSync" (for Windows XP) or "Windows Mobile Device Center" (for Windows Vista/7 systems) is present, the "MovXCESetup" folder will be automatically created in the Movicon installation folder. Within this "MovXCESetup" folder ".cab" files will be copied installing Movicn, ADO and the SQL Server Mobile on WinCE devices along with the Communication Driver ".dll" files. In addition, a "Start-All Programs-Movicon11.2-MovCE" connection group will allow you to install Movicon with all the components necessary in WinCE Devices.
The "MS ActiveSync" tool is only available for Windows XP and not for Windows Vista/7 in which you will find pre-installed in Windows Vista/7 Business the "Sync Center" as a replacement. However this replacement, "Sync Center" is not sufficient enough for the Movicon setup functions and therefore you will need to also install the "Windows Mobile Device Center" (last 6.1 version) available on the Movicon CD for the following versions:

- Windows Vista (all version except "Starter")
- Windows 7 (all version except "Home Basic")
- Windows Server 2008

In cases in where "MS ActiveSync" or "Windows Mobile Device Center" are not already present when Movicon is installed, the standard setup will not create the "MovCESetup" folder. Therefore you will have to choose "Custom" mode to carry out the setup if for the first time or if Movicon is already installed you will have to redo the set selecting "Modify" mode and enable the "Movicon-Movicon Development-MovCE Setup" item, including the sub features, until the "MovCESetup" folder is created containing the ".cab" files for installing Movicon and the relevant components for WinCE. Using this method however will not create the links in the Windows Start menu and will need you to copy and execute the right ".cab" files to the CE device/s desired to complete the installation.

When uploading a project to a WinCE device while connect with the "MS ActiveSync" plugin, a check will be made to see if the Movicin CE runtime engine and its components exists. If not you will be asked to install them.

The Movicon installer identifies which type of processor is being used by the device before automatically installing the Movicon CE executable most suitable.

You can, nevertheless, install the Movicon CE RunTime engine on the device manually by following the procedures explained below:

Installing Movicon CE using MS ActiveSync or Windows Mobile Device Center

In order to install the runtime module automatically and with simplicity on the WinCE device you will need to have the "MovCE" connection group ready for use in the Windows "Start" menu, as described above, as well as an pre-established connection between the WinCE (target) device and the PC (desktop), using "MS ActiveSync" or "Windows Mobile Device Center".
At this point from the "Start-All Programs-Movicon11.2-MovCE" menu you can continue with installing the components desired. In this case "MS ActiveSync" or "Windows Mobile Device Center" will automatically acknowledge the processor type aboard the "target" and upload the ".cab" file needed to perform the installation correctly. Once the ".cab" file has been uploaded it will be immediately executed to provide the MoviconCE RunTime modal setup or the selected components.
All the operations are done in automatic but the user is given the possibility to decide where the packet is to be installed: in the default folder for applications or in any other folder. It is advised not to use the default folder in SSDK devices as its contents will be lost when turned off because it does not store this memory.
You can select the Movicon CE installation with a "full" or "min" setup:

- **MovCE PocketPC Setup**: the Movicon CE runtime module is installed for Pocket PC devices
- **MovCE SSDK Setup**: the Movicon CE runtime module is installed for SSDK devices
- **Optional Tools**: the ADOCE and SQL Server components are installed for Pocket PC or SSDK according to what's been chosen

The Communication Drivers must be inserted in the "Drivers" folder inside the Movicon CE installation folder. In cases where the Driver Setups should create different folders all together in the device, you will need to manually copy the driver's dll to the "Drivers" folder (if it does not exist you will have to create it):

..\MovCE\Drivers

The library of the ADOCE and SQL Server Mobile components must be copied to the Movicon CE installation folder. By doing this, these files will be copied automatically to the "Windows" folder at the Movicon Startup, being a necessary condition in order for them to work correctly. When installing components with create a different folder, you will have to manually copy the files of these components to the Movicon CE folder.

Communication Driver installation

The Communication Drivers do not need any particular installation procedures. Simply copy the ".dll" files desired to the WinCE device in the Movicon CE "Drivers" folder. These files are on the PC desktop in subfolders in the "MovXCESetup" folder:

- PocketPC -> for Pocket PC devices
- StandardSDK\ARMV4I -> for WinCE panels with ARMV4I processors
- StandardSDK\x86 -> WinCE panels with x86 processors

When using the Upload/Download function to upload projects to CE devices, the TCP plugins and MSActiveSync will verify which drivers are used in the project and will transfer them to the WinCE device if not already there.

Installing the Movicon CE RunTime module manually

If 'ActiveSync has not been implemented on the "target" device by its builders, or the "MS ActiveSync" or "Windows Mobile Device Center" has not been implemented on the "desktop" PC, it will not be possible to connect to the device through the ActiveSync. Therefore you will have to manually upload the ".cab" files to the device by following these procedures:

1. Establish a connection between the "target" and the "desktop". This connection can be either network or serial type
2. Determine processor type aboard the "target"
3. On the desktop select the file with the ".CAB" extension specified for that processor from the Movicon CE installation directory, ("Programmi\Progea\Movicon11.2\MovCESetup") and copy it to the "target"
4. Double click on the CAB file on the "target" to run it. If an error arised opthe a DOS session and digit the "\>WCELOAD FileName.CAB" command
5. The Movicon CE Runtime module installation should then start. The CAB file will be cancelled automatically when the installation procedure has terminated
6. The same above described procedure will also be used for ADOCE library and SQL Server Mobile".cab" files when necessary. Therefore you should copy the ADOCE and SQL Server Mobile file to the Movicon CE installation folder as a precaution.

The SSDK devices (Touch Panel or similar) do not have all their memory buffered but only a few folders. The information which is recorded in volatile memory is therefore lost when shut off. It is recommended to execute the "RegSave" command after Movicon CE has been installed, otherwise the runtime module may not work correctly.

Uninstalling the Movicon CE RunTime Module

Windows 2000/NT and WinCE have a 'Add/Install and remove programs...' icon on their Control Panel which can be used to remove Movicon CE. However, as the WinXCE installation is customized by its makers, it is sometimes not possible to remove the programs from the Control Panel. In cases such as this, a file dedicated for removing programs called "unload.exe" in the device's Windows directory can be used. To remove Movicon CE use this command to open a DOS window and write the command:

\>unload Progea Automation MovCE

This will uninstall Movicon CE.

4.1. MS ActiveSync and Windows Mobile Device Center

The Microsoft ActiveSync and Windows Mobile Device Center programs have been provided for free in the Movicon CdRom and are used for connecting the Desktop PC to the target device in which WinCE has been installed.

By using the "MS ActiveSync" (for Windows XP) and "Windows Mobile Device Center" (for Windows Vista/7) applications, you can create a connection between the desktop PC, where the project is developed, and the target device with WinCE.
Before carrying out any connections ensure that the "Allow connection with desktop computer when device is attached" function on the target PC has been activated. The function's status can be viewed by launching the Communication program icon from the Windows CE control panel. Also double check and make sure that the "MS ActiveSync" or "Windows Mobile Device Center" application settings have been done correctly in the desktop PC: select the "Connection Settings" items from the application's File menu and enable the connection type, serial or USB, then select any necessary parameters.
The in order to establish the connection you need to launch the RepLog.exe file, located in the Windows system folder, from the target PC. The "MS ActiveSync" or "Windows Mobile Device Center" program, on the desktop PC, will automatically establish a connection, otherwise you can force it by launching the "Get Connected..." command from the ActiveSync program's File menu. When opting for the USB connection you will need to executed the "Repllog.exe" file on the WinCE device.
The desktop and target PC connection allows Movicon projects to be exported directly in Windows CE, new program installations on the target PC or to explore the disk's contents. The connection status can be viewed in the Windows application status bar where the status type is indicated the color of the "MS ActiveSync" or "Windows Mobile Device Center" icon: green means the connection is active and grey means it is disconnected.

Each time the cable is disconnected, you need to repeat the procedure described above to reconnect.

Communicating with MS ActiveSync or Windows Mobile Device Center

The Movicon packet for Windows CE installation requires that the desktop and target are communicating with each other. The connection is established thanks to the MsActive Sync (at least version 3.5) and comes in three types:

1. Serial or Infrared communication through a COM Port
2. Comunication through USB
3. Network (Ethernet) Communication and with Remote Access Service (RAS)

As soon as communication between the desktop on target has been established, the MS ActiveSync or Windows Mobile Device Center will ask if you want to setup a partnership.

The partnership is needed when you wish to share and maintain syncronized data between the desktop and target. Normally the partnership is frequently used in Pocket PC devices but not in SSDK devices where communication is not always carried out successfully due to the lack of adequate modules in the target device.
The SSDK devices are generally designed for in the industrial world. Different types and sizes can be found on the market: some have communication serial ports between 232 and 485, ethernet cards, infrared ports and USB ports.
However, PocketPCs are directed at a wider public use due to their extreme simplicity and user friendliness, and are capable of quick data sharing with the desktop such as: phonebooks, internet addresses, files and others still.

Installing MsActiveSync

The "MS ActiveSync" and "Windows Mobile Device Center" applications are free Microsoft software designed for managing communications between PC Win32/64 (desktop) and WinCE (target) devices. The "MS ActiveSync" tool is only available for Windows XP and not for Windows Vista/7. Windows Vista/7 replaces this tool with the "Sync Center" which has been pre-installed in Windows Vista/7 Business but, however, is insufficient for the Movicon Setup. Therefore you will also need to install the "Windows Mobile Device Center" (last 6.1 version) available in the Movicon CD for the following versions:

- Windows Vista (all version except "Starter")
- Windows 7 (all version except "Home Basic")
- Windows Server 2008

If "MS ActiveSync" or "Windows Mobile Device Center" has not already been installed on your desktop PC, you can easily install them from the Movicon installation CD for free.

4.1.1. Serial Connections

Serial Communication with MsActiveSync or Windows Mobile Device Center

The serial communication works in all Windows CE devices and you only need to have a free serial port on your PC, a standard serial cable and the Active Sync or Windows Mobile Device Center installed on the Desktop. The communication velocity can go up to 115200 baud rate.
To get serial communication you need to carry out the following steps:

1. Install Active Sync or Windows Mobile Device Center on the Desktop. Once installed select the COM port you want to use. Active Sync will remain active, on hold, in wait for serial signal reception. ATTENTION: keep the serial port engaged even when no Active Sync communications are active. To free the serial you need to enable the connection option to the serial which is accessed from the "File-Connections setting.." menu
2. Open the control panel on the Target device and access the "Communication" settings
3. Enable the "Allow connection with Desktop computer when device is connected" selection in the "PC Connection" properties
4. Always check that the connection being used is correct in the "PC Connection" properties. You can always create new communications afterwards by accessing "Remote Networking" folder from the Programs/Communication
5. Close the Communication Settings window with OK. Close the control Panel
6. Connect the Desktop and Target device with a standard serial cable
7. Run the executable "repllog.exe" by selecting Run item from the Windows CE start menu accessed from the task bar
8. The communication will activate automatically

9. A window will open, on the Active Sync or Windows Mobile Device Center Desktop, enquiring whether to activate a "Partnership". It is not necessary to activate a partnership of the Movicon functions
10. The serial communication is now active

By using this type of connection you can transfer files between two devices, the Desktop and the Target. The "Mobile Device" can be displayed by using the Desktop's explorer to which you can access for reading or writing files. In addition to this you can export Movicon projects directly to the Target without having to "copy&paste". After having created a Movicon project you can actually execute the "Upload Project to Device/FTP..." from the"Commands" windows of the "Project Explorer" window. If the connection between the Desktop and the Target has already been activated by using one of the plugins, Movicon will directly export the project to the Target, keeping the same .movprj source file path.

Communication on USB port

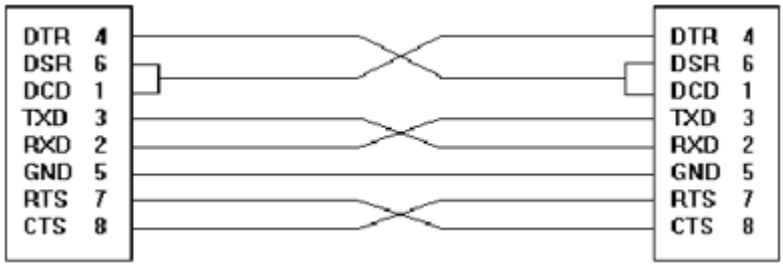
The communication on USB port is usually available in PocketPCs and in almost all of the SSDK devices because it is the simplest type to put into effect. In fact, you only have to connect cable to the PC's USB port and wait until the communication is established automatically.

4.1.2. Serial Cable for MS ActiveSync and Windows Mobile Device Center

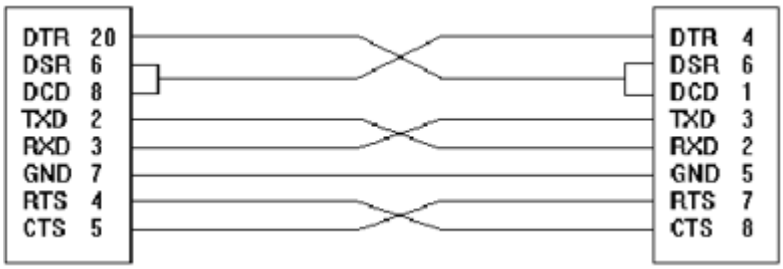
The communication serial cable to be used for connecting the desktop PC to the target device using MS ActiveSync or Windows Mobile Device Center is as follows:

Nevertheless, it is always best to refer to the device's manufacturer's specifications if available.

Serial Cable with 9 pin (female) - 9 pin (female) connector



Serial Cable with 25 pin (male) - 9 pin (female) connector



4.1.3. Network Connections

Network communicating with Ms ActiveSync

Network communication using Ms ActiveSync is only available for MS ActiveSync versions up to 3.5 and requires that both PCs, Desktop and Target, be supplied with a network card and be configured adequately. This system is used for transferring files at a much higher speed than a serial

communication does with a maximum baud rate of 115.2 Kb. Communicating in net can be established only after a partnership has been created, which can only be established if there is a communication serial or USB port to start with.
The procedure for creating a network connection is as follows:

1. Connect the CE device up to the desktop with a serial cable
2. Connect the CE device to the same PC's network
3. If the net is without the DHCP service, you need to setup a fixed IP in the device which is compatible with that of the PC's. When the settings have been completed you need to save the registry to keep it intact if any reboots are carried out afterwards
4. Setup the ActiveSync on the desktop to accept serial, USB and Network type communications
5. Launch "repllog.exe" on the device
6. Establish a partnership with the device. Disable all the check boxes in the objects for partnership managing (they are not necessary). Information concerning the Desktop PC is memorized in the device's register while information concerning the device is recorded in the desktop PC's register. Therefore the register needs to be saved to keep information intact after a reboot
7. Disconnect the serial connection by using the appropriate repllog icon from the Windows CE task bar
8. Launch "repllog.exe /remote"
9. A window will open indicating the name of the desktop PC with which the partnership has been established. Click the 'Connect' key
10. The connection is put into effect via ethernet with the desktop by means of using the ActiveSync

If the desktop PC has not been changed, all following connections can be done directly in ethernet even when the desktop or device has been turned off and turned on again. Otherwise when using another desktop PC you will need to re-establish the connections via serial with the partnership again.

Network Communication via FTP Server

Network communication via FTP Server requires that both the PCs, Desktop and Target, be supplied a network card and be configured adequately. This system is used for transferring files at a much higher speed than a serial communication does with a maximum baud rate of 115.2 Kb.
In addition to this a FTP server must be installed on the CE device.
Progea has created a FTP Server which is loaded on the device during Movicon CE installation. The project can be uploaded by using the "Upload Project" window once the FTP Server is put into execution on the CE device. You will need to select the "Server FTP" option from the "Project Upload" window and specify the CE device's IP address and its network name. However, this connection is only used for uploading the project the CE device. It cannot be used for executing the Movicon CE installation or starting up the project from the Desktop PC or creating connection in the Target device.

5. Editing

5.1. Editing General Information

When creating projects for the Windows CE you need to take into account the differences there are compared to the Win32/64 platform version. The main difference is that the devices taken on board Windows CE are divided into two different categories:

- **Pocket-PC**
- **SSDK or HPC2000**

The Pocket-PC, also called palmtops, are rather small and have RAM memory storage. As you can imagine by its name, these devices are mobile and not much bigger than a mobile phone.
The SSDK devices (standard software development kit) are similar to operator panels and generally used in the industrial environment. They come in different sizes according to their display type, which can oscillate between 5.7" and 15". These devices have a RAM memory and a Compact-Flash memory only, which carries out the tasks of the Hard-Disk. The Operating System's image resides in the Compact-Flash and is loaded into RAM at device startup.

Windows CE Operating System

The Windows CE platform is rather different to the Win32/64 platform as described below. You must first consider that WindowCE, like embedded systems, is much more restricted both on the performance side and with the amount of memory space available.

Features

In order to verify some of the device's features, you can access the "System" group from the "Control Panel" to get information on the following:

- Operating System and relative version
- Processor type
- Installed RAM memory
- Memory management: Storage Memory, being the memory reserved for system files; Program Memory, being memory available for running various applications
- Device name and description

To get further information on the device please refer to the section on "Project Debug".

Connections

The CE devices are connected to the PC desktop using serial , USB or network connections. Normally each device proposes a serial connection for default, such as ActiveSync, which is provided for the connection. The available connections are listed in the "Network and Dial-Up Connections", found on the "Control Panel", where you can also create new connections. However, please keep in mind that the serial connection via ActiveSync is normally executed on CE device's default connection (My Connection). In this case you need to check that the connection in question is that used by the device otherwise it can be setup through the "PC Connection" found on the "Control Panel". The window which appears shows which is the connection currently selected, and the following item needs to be enabled:

Allow connection with desktop computer when device is attached

Where the SSDK devices are concerned, you need to use the serial or USB connection the first time you execute a connection with the Desktop PC, after which you can then connect in net as explained in the paragraph on "Network Connections".

Registry

The device's system registry is modified by the Movicon CE installation, where all the keys needed for making Movicon CE work are created. Therefore it is absolutely necessary that the system registry is

saved after each change has been made. However, there are usually no problems when it comes to Pocket-PCs which stores its memory so that there are no risks of losing data when the device is turned off. The SSDK does not store its RAM, therefore when the device is turned on again, any system registry changes will be lost if not saved in the device's Compact Flash before being turned off. If saved, the system registry will be loaded along with the last changes when the device is next turned on. The system registry commands are different from one device to another, therefore the system registry save will have to be executed according to the device being used.

Pocket-PC devices are supplied with a battery to keep data always saved in memory. However, when the batteries run out or get damaged all the saved data will be lost. Therefore it would be wise to regularly do a backup copy of the memory as a safeguard against losing data.

DOS Command Window

The DOS command window is opened by means of the "Command Prompt" item from the Windows CE Start menu. There are only a few commands provided which are listed by clicking the "Help" command. These commands include "ipconfig", "ping", etc..

Accessing the Compact Flash

Access to the Compact Flash is rather slow especially with SSDK devices. You may find that it takes time to open a certain folder to view its contents even when using the explorer resource. This usually happens when the MocXCE installation folder is being opened because the Movicon icon, to be associated to the "movprj" project file, is loaded.

Project compatibility between desktop and target

One of the biggest advantages of using Movicon is that you get a XML based project structure. Which means all the same project files run on Win32/64 can be uploaded to the device and run by the Movicon CE runtime engine.
As a consequence, there is no need to compile the project file when uploading it from Desktop PC to CE device and project reverse engineering is no longer necessary.
This makes project handling extremely more simple.

Project Options

You can setup the project to be run on full screen or no by means of using the project's "Execution" settings. If you choose to activate the "Start Full Screen" option it would be advisable to insert a command (by button or by menu) to close the project. If the CE device does not have a keyboard, **it will not be possible to startup the operating system** when run on full screen making it impossible to stop the project without having to turn the device off.
The project's Output window can also be enabled but also in this case if the project is run on full screen, it will be impossible to access the window without using the keyboard (ALT+TAB keys and then select the process).

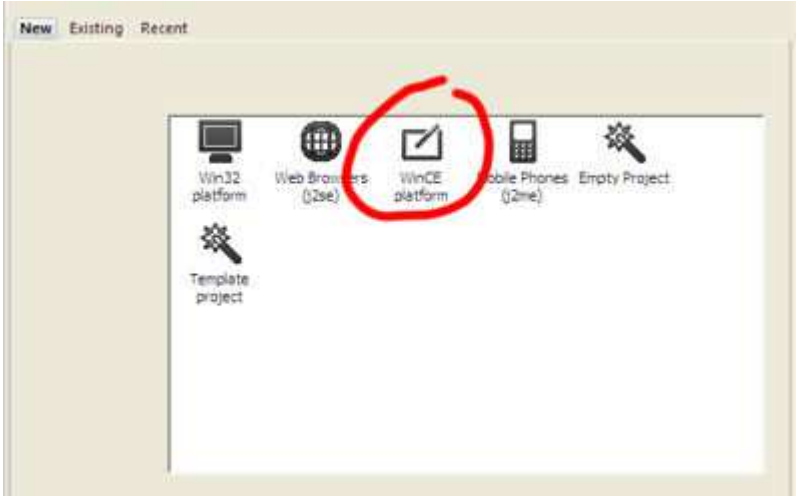
Folder Paths

Project working folders can be setup through the project's "Project Path" settings. You must take into account that the "C:\" disk does not exist in CE devises therefore the "C": is automatically taken away by Movicon. The working folders are used, above all, for defining the folder images and when Dataloggers and Historical Logs are to recorded on a compact flash which is different from the one where the Movicon installation resides.

5.2. Designing Projects

Creating New Projects for WinCE

By using the Movicon Editor, you can create a new project by selecting the target device's icon as indicated below:



Although projects created with Movicon are executable WinCE devices as well, it would be best to create the new project by selecting the target device: this will automatically and optimally pre-configure the project's properties for WinCE devices.

Selecting platforms in Design mode

When creating a project for Windows CE you will need to consider the following restrictions and differences between using Windows 32/64 bit and Windows CE, which involve the:

1. Significant hardware restrictions
2. Operating system restrictions
3. Consequent restrictions in Movicon CE

Creating a new project in Movicon for WinCE will automatically set the project's **"General properties"** in the **"Platform"** group. However, these properties can always be changed to set the platform type being designed with the Movicon Editor.

- **The Movicon Editor will hide those functions not supported by the target according to the platform type selected and display some graphic objects exactly as they will be represented in the device.**

To modify the project's **Platform** properties, select the project root with the mouse and use the Movicon **"Properties Windows".**

Note: when designing for Windows CE, **it's advisable to create a new project explicitly for Windows CE to get all the default settings configured correctly!**
However, Movicon will accept more than one platform selection in the project's properties and assumes that different platforms are to be designed **for which the editor will always be preset for the most powerful one.**

Also take into consideration that when transferring projects designed for Win32/64 to WinCE devices, **the Movicon CE runtime will ignore and not manage all those functions not supported which will be notified** in the Output window and in the Log upon project transferral. Even though the project will run in the WinCE device, the designer should check whether it does so correctly keeping a special eye on performances and memory consumption.
Let it be known that all Movicon X projects, being based on XML files, **can be run on both Win32/64 and WinCE indifferently**.

Before Designing

Before starting your project for embedded devices you should always remember that your project must be created in parity with the capabilities and possibilities of your target device.
Devices based on WinCE are generally compact devices with restricted calculation and memory capacities and therefore we strongly suggest the designer to keep in mind all the tips mentioned in this guide and in the **"GuideLines for WinCE Touch Panels"** manual.

5.3. Configuration File

Movicon CE does not use the Windows registry keys for reading attributes which consent to modifying some of its working features. Instead it used the "MovCE.ini" file which must be found in the Movicon CE installation folder. The MovCE.ini file has a XML root tag called "Settings" in which all the Movicon sub keys with their respective values are inserted. This file is structured as follows:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<Settings>
<General>
<DontUseDecorativeFont>1</DontUseDecorativeFont>
<InstallComponents>0</InstallComponents>
</General>
<MouseCursor>
<Visible>0</Visible>
</MouseCursor>
</Settings>
```

In this example you can see that two values have been inserted in the "General" key and one value has been inserted in the "MouseCursor" key.

Caution! The MovCE.ini file structure has changed in respect to the previous Movicon versions, and not compatible with the MovCE.ini files created for the previous versions.

Keys available only for Movicon CE

Key	Value	Description	Default
General	CheckFileTimeAndDate	This key is used for enabling/disabling the control of file data while exporting projects to WinCE.	DWORD = 1 0 = Disabled 1 = Enabled
General	ShowSIP	This key is available for WinCE only. It allows the WinCE virtual on screen keyboard to be deactivated. When set at zero the WindowsCE on screen keyboard cannot be activated.	DWORD = 1 0 = Disabled 1 = Enabled
General	ADOCEProvider	This setting identifies the Provider used for the connection to the Database when using Movicon CE.	String = Microsoft.SQLSERVER.MOBILE.OLEDB.3.0
General	ADOCEDataSourceExt	This setting identifies the extension used or creating DataBase files when using Movicon CE.	String = .sdf
General	ILLogicPriority	This value expresses the priority with which the IL logic is to be run.	DWORD =255
General	MaxAvailVirtual	This value expresses the percentage of memory in use that when exceeded	DWORD = 5000000 Bytes

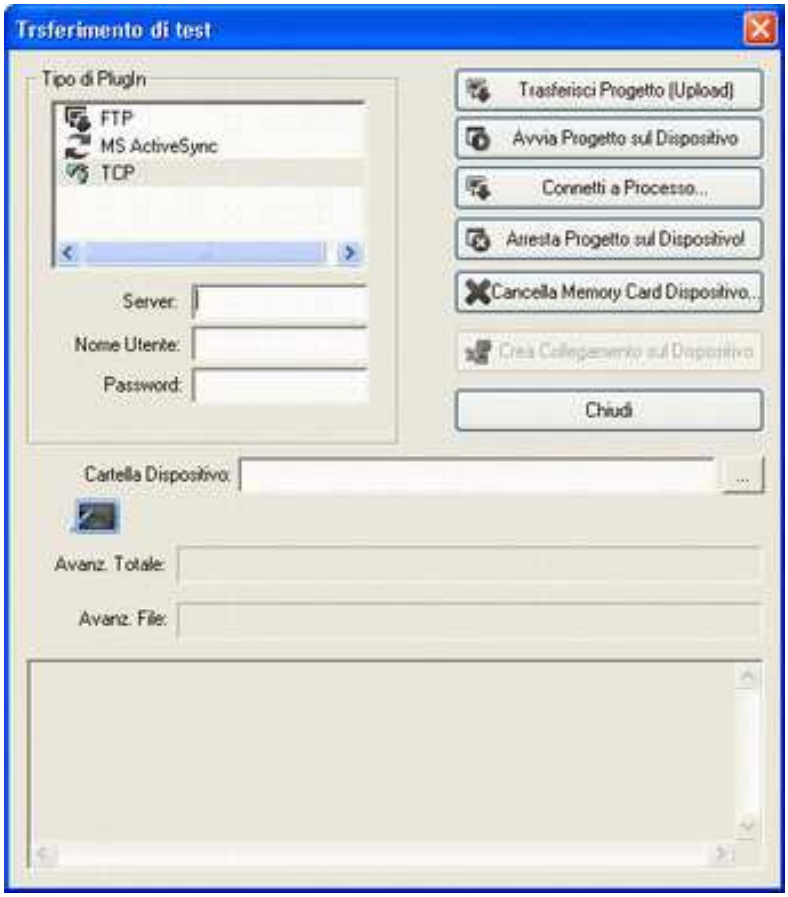
		the situation of the screens, which are in memory but not displayed, becomes critic for which they will be unloaded from memory creating a message in the trace. Unloading is done independently from the screens' active 'Not Destroyable' option. Furthermore the change page will be managed as if the screen property "Close Screen Delay" is set at "0". This management is disabled when set at "0" value.	
General	InstallComponents	This key is valid only for WinCE. Its default value is "1" and, when set at "0", allows the installation of the ADOCE and SQL Server CE components to be deactivated. It is used, for instance, to lighten the memory load in those devices where the ADOCE and SQL Server CE has been pre-installed and in the project to be run where they are not needed. Handling data from the IMDB becomes lighter and quicker with the same potentialies interms of queries.	DWORD = 1
General	DontUseDecorativeFont	Some font properties are no longer loaded for default in Movicon CE which require a high usage of resources. This will increase system performances but may make the fonts less attractive to look at. The loading of fonts can be restored to their original mode by setting this registry key at zero.	DWORD = 1
General	MaxCacheFont	This key allows you to use a cache for storing fonts used by MoivXCE so that they don't have to be recreated every time they are needed. The cache is set with a maximum limit to stop the system from being saturated with fonts when there are many different types.	DWORD = 0

General	CompactAndRepairEnabled	This key allows you to perform SSCE 3.0 database compact and repair at the project startup if needed. The "1" value enables the automatic compact. When the compact has been enabled, the Movicon CE startup in the historical log initialization phase changes its extension from ".sdf" to ".bak", compacts, and repairs it and the creates a new file with the ".sdf" extension. If compacting fails the previous historical log is restored changing back from the ".bak" to the ".sdf" extension. If you don't compact at startup, you will save about 750 KB of physical memory space, save space on the compact flash (depending on number and size of historicals) and the project startup becomes quicker. On the other hand, when compacting and repairing at project startup, this will allow the resetting of any errors in the historical logs and re-organize database indexes for quicker data accessing when in large amounts.	DWORD = 1
MouseCursor	Visible	This key enables/disables mouse cursor visibility when the application is being run. The "1" value makes the cursor visible and the "0" value makes it invisible.	DWORD = 1
General	UseMouseGestureOnScreen	When this value is set at "1" the "mouse gesture" is managed in screens in WinCE. This means that when you keep the button moused pressed down on a screen area without objects, a menu will appear allowing the following commands to be executed: • Zoom In • Zoom Out • Zoom To • Reset Zoom • Refresh	DWORD = 0

5.4. Project Upload/Download

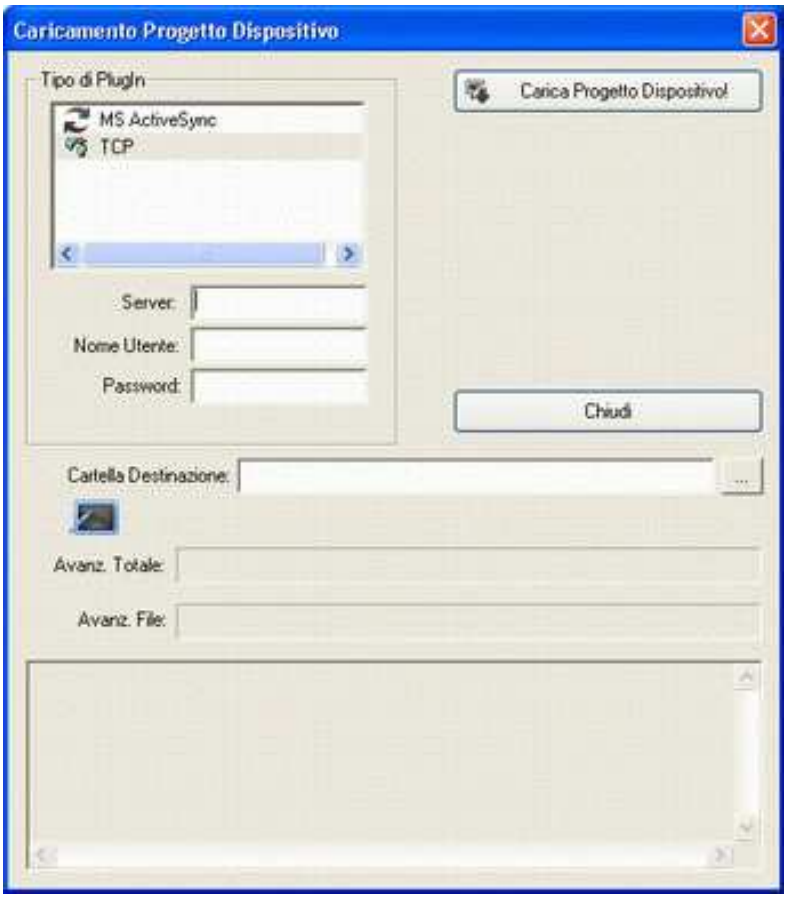
Uploading/Downloading Projects

Movicon projects can be uploaded to WinCE or Win32/64 devices using the "Upload" window accessible through the **"Upload Project to Device/FTP..."** command found in the "Tool Bar" or from the menu which appears right clicking on the project name in the Project Explorer Window. This is how the Upload window will appear:



The project upload to device operation can be performed using the different protocols or plugins: FTP, TCP and MS ActiveSync.

Projects can also be downloaded from WinCE or Win32/64 devices in the same way, using the "Download" window accessible through the **"Download and Open Device pro..."** command from the "Tool Bar" or menu activated with a right click on the project name in the Project Explorer Window. The Download window will look like this:



The download project from device operation can be done by using the different protocols or plugins: TCP and MS ActiveSync. The (MovCE.boot) boot file is used for performing this operation. The project that will be taken from the device is actually the one defined in the boot file.

The Upload/Download project window command buttons and entry fields are described below:

"Upload Project" command

This command uploads project to the device using the destination path specified in the "Upload Device Path" field. This destination path is optional. If this parameter is set, before starting the upload the user will be asked to select the destination compact flash from a list (there may be more than one compact flash on the device). At this point the project will be uploaded rebuilding the same local PC desktop path in the selected compact flash as well. The path to use for uploading the project can also be selected using the browse window ("..." button to right of "Download Local Path" field) which will show all the folders and sub folders currently in the device.
In cases with devices with "No Root Mount" images, compact flashes will not be shown in the operating system with a specified name. For these devices the exportation procedure is performed in the \MyDocuments folder unless a destination folder has been indicated. Boot files are also created in the \MyDocuments folder where the application is loaded.
The boot file will be created in the compact flash root upon upload termination so that the project will be loaded and run when the application is next started up. If no startup screen has been set in the project (Going into Runtime mode from the development mode always opens the screen being worked on at that moment and if setting up a startup screen has been forgotten), a warning message will appear requesting confirmation to proceed with project uploading.
When using the TCP plugin, the destination device's date/time is synchronized with the source desktop PC before download begins. To speed up this process only those project files modified since the last transfer will be transferred.
The date/time update and seletive transfer functionalities are active if the source desktop PC has the HKEY_CURRENT_USER\Software\Progea Automation 11.2\MOVICON\ExportCE\CheckFileTimeAndDate registry key set to value. Date/time will not be sychronized if PC does not have this key, or has key set to value 0 and all the project files will be tranferred.

In cases where the transferred project uses communication drivers, a check is run for drivers in the destination device and on the version if found. Then a request will be made whether drivers are to be copied or updated on device. The drivers are found in the Source PC's MovXCESetup subfolder, divided by device type. If the Windows CE configuration registry on the destination device has the HKEY_LOCAL_MACHINE\SOFTWARE\Apps\Progea Automation MovXCE' key, the path specified in the 'InstallDir'value for installing the driver in the 'Drivers' subfolder. On the contrary, drivers will be in stall in the 'Drivers' subfolder relating to the folder in which the CEUploadServer tool resides.

Using the MS ActiveSync protocol before starting the download procedures synchronizes the date/time of the destination device based on the data/time of the source PC desktop. To make this procedure faster, only those project files modified since the last download will we downloaded.
The date/time update and selective download features are active only when the following conditions have been verified:

1. the HKEY_CURRENT_USER\Software\Progea Automation 11.2\MOVICON\ExportCE\CheckFileTimeAndDate registry key with value 1 is in the PC desktop source. If this key is not present, or set with the 0 value, the date/time will not be synchronized and all project files will always be downloaded.
2. the CERapiCDate.dll library is in the path specified in the "HKEY_LOCAL_MACHINE\SOFTWARE\Apps\Progea Automation MovXCE" registry key's "InstallDir" value in the destination panel.

when using the MS ActiveSync protocol, a check is made in the destination device to see if any communication drivers are being used in the project. This search is carried out in the 'Drivers' sub-folder of the path specified in the "HKEY_LOCAL_MACHINE\SOFTWARE\Apps\Progea Automation MovXCE" registry key's "InstallDir" value. No control checks are made if this key is not present. If there aren't any drivers present when the control check is being made, a request will be made to indicate whether device is standard type (ARMV4I or x86), or PocketPC and then the drivers will be copied from the PC source to the 'Drivers' sub-folder of the path specified in the 'InstallDir'.

When the project download window is closed the settings of the Plug-in being used will be saved in the "<ProjectName>.upldsettings" file in the project's "DATA" folder, so the next time the window is opened, the Plug-in's parameters will be re-read from this file. In this way each project keeps its own Plug-in settings.
The following information is saved in the file:

PlugIn used (Type): 0 = MS ActiveSync; 1 = FTP, 2 = TCP
IP Device (Server): device's IP address where project will be downloaded to
User Name (User): name of authentication user
Device Folder (Path): destination folder of project in device.

"Get Project from Device!" command

This command is found in the download window and is used for retrieving projects from a device and copying them locally. The project will be copied in the folder defined in the "Download local path" field. If nothing is set in this field a browse window will open for selecting the local path in which to save the project. Project downloading operation uses the boot file found in the device for singling out the last project uploaded to the device (which may be running). The command is aborted if the boot file is not found in a compact flash root.

"Start Device Project" command

This command first arrest any project running in the device, then starts up the last uploaded project, being the one set in the boot file. An error message will be generated, if there is no boot file or if it does not contain a valid project.
This command can only be used when connected to the device through the TCP or MS ActiveSync plugin.

"Attach to Process..." command

Once the project has been uploaded to the device and put into run mode, it can be connected to using the "Attach to Process..." command to perform project debugs. This will allow you to display variable values, perform script debugs, etc... For further information about this function please refer to the section on "Remote Project Debugs" in the Movicon programming guide.

"Stop Device Project" command

This command stops any project running on the device.

This command is only possible when connected to the device through the TCP or MS ActiveSync plugin.

"Erase Device Memory Card" Command

This command deletes all the contents of the folder specified in the "Upload Device Path", and allows you to empty the compact flash without accessing the panel directly.

All the files included in the specified path will be deleted without controlling file type. Therefore each time this command is invoked you will be asked to confirm in order for this operation to proceed. cancellation before this procedure goes ahead.

"Create Device Shortcut" Command

Once the project has been uploaded to the device you can create a shortcut to the device in order to run it. The "Create Device Shortcut" button creates a link in the device's Main Memory, through which the project is put into run mode.

Movicon creates this shortcut to the project in the device's Main Memory. This means that the link will be lost when the device is restarted and for which we advice you to copy it in the Compact Flash.

Server:

Name or IP address of the Server to be connected to.
This parameter is only available for the FTP and TCP plugins. The port in addition to the IP (Es. "192.168.0.10:6000") address can only be specified in the TCP plugin. This may be handy when for example the panel does not have a public IP address and the packet must pass through a router.

User Name:

User name for Server connection authentication.
This parameter is only available for the FTP and TCP plugins.

Password:

Password for Server connection authentication.
This parameter is only available for the FTP and TCP plugins.

Upload Device Path/Download Local Path

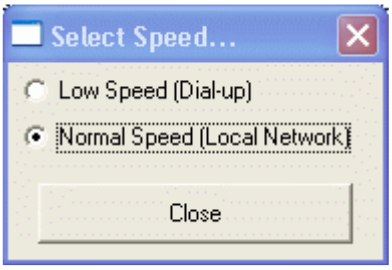
This field represents the destination folder in which the project will be up/down loaded to. This folder is either from the remote device or local PC according to the Upload or download operation performed. The folder can also be selected by using the browser button to the right.

FTP Plug-in

The FTP protocol can only be used for uploading the project to devices which have an active FTP Server. Therefore the "Get Project from Device", "Start Device Project", "Stop Device Project", "Attach to Process...", "Erase Device Memory card" and "Create Device Shortcut" command are not supported. If the device does not have a FTP Server, you can use Movicon's. The Movicon CE installation includes the "FtpSvr.exe" file inside its installation folder which can be started up just before connecting to the device.

TCP Plug-in

The TCP protocol can be used for uploading and downloading projects to and from devices with Windows CE or Windows 32/64 bit. In order to use this protocol you will need to start the "CEUploadServer.exe" application (Window CE devices) or the "TCPUploadServer.exe" application (Windows 32/64 bit devices) up on the device beforehand in addition to having a valid Ethernet connection. These two files are installed with Movicon installation, WinCE and Win32/64 versions, and are found in the same Movicon executable folder. In cases where you use the TCP plugin, first make sure that the UploadServer application has been startup on the device. The TCP plugin allows project uploading with slow connections as well (such as RAS connections via modem).
The first time the plugin is loaded it will request you to specify connection speed used ("Normal Speed" for default). The other options are "Low Speed" and "Normal Speed". Slow speed is adaptable for remote connections with low band rates, which exploit Dial-Up type modems.



The TCP plugin will communicate to the remote server which connections to use in function with those selected, while the timeout and buffer size parameters will be taken from the "CETransfer.Settings" xml file.

When the source PC and the destination PC both use Windows XP/Vista/7 as operating systems, the project 'upload/download using the TCP plugin will request authentication for destination PC. Therefore you will need to insert a password and user name known as destination PC administrator user in the Upload and Download window's "User Name" and "Password" fields. The User in question may be a local machine user or a dominion user belonging to the same machine dominion In both cases, however, the user password must not be left blank.

If you try to Upload or Download from a source PC with a plugin which has not been updated (CETransfer.dll file with version prior to 1.2.0.7) towards a destination PC with an updated plugin, this procedure will fail to work. This message will appear in the log window: "The upload/download (CETransfer.dll) manager is not updated in this computer!"
No authentication is required if destination device is a WinCE device.

Transferring one project at a time

When using the TCP Plug-in you can transfer one project at a time. Projects on destination device are overwritten by those being downloaded if different without requiring user authentication. To obtain this feature you will need to set the "CETransfer.Settings" configuration file's "ProjectPath" parameter described in the next paragraph. The project's destination folder should be specified in the "ProjectPath" parameter (i.e. "ProjectFolder" or "Projects\Movicon", etc.) and the plug-in will transfer the project to this folder. In this case when browsing device it will possible to select the project's memory card or destination disk unit only but not any folder. In cases where a path is entered manually in the "Device Folder" field, this wil be ignored by the Plug-in. Once the new project has been transferred the boot file will then be updated as well.
Setting the "ProjectPath" parameter with a '0' value will disable the transfer one project at a time management.

When the transfer one project at a time feature is active, when downloading a project that is different to the one on the device, or memory card or destination disk unit has been changed, the old project will be deleted. In this case the deleted project will be the one defined in the ".boot" file.

"CETransfer.Settings"file description

This XML file contains settings that are used by the TCP plugin on client and server side. It is recommended to leave these settings as they are unless indicated otherwise by technical support.
The "CETransfer.Settings" file settings are loaded only at "CEUploadServer.exe" or "TCPUploadServer.exe" tool startup, therefore any changes made to this file will require "CEUploadServer.exe" or "TCPUploadServer.exe" tool restartup to acquire these changes.

"CETransfer.Settings" file description

This XML file contains the setting which are used by the TCP plugin on client and server side. We advice you not to modify these settings if not explicitly requested by technical support.

IPAddress: Only used by the server and shows the last IP address used and expressed with the "xxx.xxx.xxx.xxx" format. It is no longer used by the 1.1.0.1 version (CETransfer.dll file) because this parameter is passed by the VBA code which manages the TCP plugin.

PortToConnect: Used by client to open a socket port in listening mode and is used by the server in order to know which socket port it must connect to (default "10651").

ServerPort: Used they the client in order to know which server socket port it must connect to and is used by the server to open a socket port in listening mode (default "10651").

IdleTO: Idle time expressed in microseconds for handshaking operations between the client and server (Default "300").

ReceiveFileTO: File reception timeout expressed in microseconds (default "500000"). Amounting to "ReceiveFileTOsec" defining the total reception timeout when fast or normal connections are used.

ReceiveFileTOsec: File reception timeout expressed in seconds (default "2"). Amounting to "ReceiveFileTO" defining the total reception timeout when fast or normal connections are used.

LowReceiveFileTO: File reception timeout expressed in microseconds (default "500000"). Amounting to "LowReceiveFileTOsec" defining the total reception timeout when using slow connections.

LowReceiveFileTOsec: File reception timeout expressed in seconds (default "10"). Amounting to "LowReceiveFileTO" defining the total reception timeout when using slow connections.

LowSpeedBuffer: Working Buffer size for slow connections in bytes (default "512"). The value for using fast or normal connections is 8 KBytes.

HighSpeed: default value of the connection type to be set (slow or fast) used by the client. The default value is "1" which indicates a fast connection. This is no longer used by the 1.1.0.1 version (CETransfer.dll file), because this parameter is passed by the VBA code which manages the TCP plugin.

DevicePath: is used for specifying the default memory card in which the project will be downloaded. This parameter is only considered if the destination device uses the WindowCE platform.

ProjectPath: this parameter is used for specifying the destination path where project will be downloaded. If this parameter is set with a '0' value, as well as defined with the project's destination folder, the management will also activate according to which project can be downloaded at a time on the panel. This means that a project that is different to the one on the panel will be downloaded, therefore the later will be deleted and replaced with the new one for downloading. this operation will need user confirmation.

"CETransfer.prd" file description

This file is in both the development and runtime environments and contains a complete range of parameters in XML format as described below:

SettingsPath: Default path where project download is performed. This is no longer used by the 1.1.0.1 version, because this parameter is passed by the VBA code which manages the TCP plugin.

ExeName: Name of executable file to be started or stopped upon Start/Stop project command in the Desktop environment (Default "MoviconRunTime.exe").

Switch: Eventual parameters to be inserted into command line of executable file specified in the "ExeName" (default "") parameter. For instance, the "/R" can be used.

CEShutExeName: Name of executable file to be started upon Stop project command in the WinCE (Default "ShutXCE.exe") environment.

CEExeName: Name of executable file to be started upon Start project command in the WinCE (Default "MovCE.exe") environment.

CEBootName: Name of boot file to be created after project upload to WinCE (Default "\\MovCE.boot").

CERegistryKey: WinCE configuration Registry path in which the keys are found for getting the positions of the executable files to be started (Default "SOFTWARE\\Apps\\Progea Automation MovCE").

"TCPUploadServer.exe" application description

This executable file is installed with Movicon and allows the service to be used on desktop (Win32/64) platforms for managing project upload/downloads from remote client stations. What happens when using servizio Windows 32/64 bit service:

- The "Create Shortcut Device" command creates a connection and saved it in the PC desktop

- The boot file is created in the root of the disk used for uploading project and is used only when retrieving the project with the "Get Device Project" command, whereas the "Start Device Project" command runs the application with the project uploaded to the device just before or, if nothing has been uploaded, runs the last project which was uploaded/run on the device

MS ActiveSync

The MS ActiveSync protocol can be used for uploading projects to the device or downloading projects from Windows CE devices. It needs an open ActiveSync connection to the remote device (serial, USB or Ethernet).

There are a few values available for the "Platform" key which can be inserted in the Movicon Configuration File or in the Windows registry (see section on the "Movicon Configuration Keys) which allow you to customize some of the aspects concerning the project "Upload/Download" management when the MS ActiveSync plugin is used:

Platform\DeviceInstallPath: Path used for searching for the "InstallDir" value in the Windows CE configuration registry (default = 'SOFTWARE\Apps\Progea Automation MovCE'). The "InstallDir" value contains the path in which the runtime has been installed and is created by the project installation pack. It is used upon project Start/Stop command.

Platform\DeviceExecutable: Name of runtime application installed in the Windows CE device (default = 'MovCE.exe'). It is used upon Start/Stop command.

Platform\LocalDriverDeviceSetup: Path relating to the search for files for setting up communication drivers (default = 'MovCESetup').

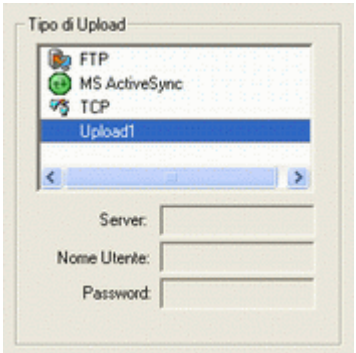
Platform\LocalPocketPCSetupPath: Path relating to the search for files for setting up PocketPC platform applications (default = MovCESetup\MovCE.PPC.ini'). It is used when the project Start command fails or to find the product's installation cab file for Windows CE pocketPC platforms.

Platform\LocalHPCDeviceSetup: Path relating to the search of files for setting up application for the platform SDK standard (default = MovCESetup\MovCE.SSDK.ini'). Used when the project Start command fails and finds the products installation cab file for Windows CE platfrom SDK Standard.

Platform\DeviceBootFile: Name of boot file (default = '\MovCE.boot'). Used after project upload for creating the boot file and used in the project download during the boot file search.

5.5. Creating Custom Plugins

The project upload/download management also consents custom plugins which can be created using script code (which will call the dll and/or native code). The upload/download user interface window displays the list of available plugins. Any further plugins will be added to this list in addition to the already existing "Ms ActiveSync", "FTP" and "TCP" plugins.



These plugins must be installed in a Movcion sub folder headed "UpdMgr". Movicon will search for all the file contents in this folder with the ".updmgr" extension and display them in the list using the file name and any eventual <nome plugin>.ico icon.

The custom plugin file must contain the basic code with the following event definitions:

```
Sub OnLoading(Output As String, bRet As Boolean)
Sub OnUnloading()
Sub OnConnect(ServerName As String, UserName As String, Password As String,
TargetFolder As String, Output As String, bUpload As Boolean, bRet As Boolean)
Sub OnDisconnect(Output As String)
Sub OnCreateDir(DirName As String, Output As String, bRet As Boolean)
Sub OnCopyFile(Source As String, Dest As String, Output As String, bRet As
Boolean)
Sub OnDeleteFile(DeleteFile As String, Output As String, bRet As Boolean)
Sub OnStartStop(bStart As Boolean, Project As String, Output As String)
Sub OnCreateShortcut(Project As String, Output As String)
Sub OnWinVersionInfo(dwMajorVersion As Long, dwMinorVersion As Long,
dwBuildNumber As Long, dwPlatformId As Long, szCSDVersion As String, bRet
As Boolean)
Sub OnGetFolders(Path As String, FolderList As String, Output As String, bRet As
Boolean)
Sub OnCleanMemoryCard(Path As String, Output As String, TotalNumber As
Long, bContinue As Boolean, bRet As Boolean)
Sub OnGetBootFile(BootFile As String, DevicePath As String, TotalNumber As
Long, Output As String, bRet As Boolean)
Sub OnDownloadFile(Path As String, LastFile As String, Output As String,
bContinue As Boolean, bRet As Boolean)
Sub OnGetDeviceInfo(dwPlatform As Long, bIsPocketPC As Boolean, Output As
String, bRet As Boolean)
Sub OnCheckDriver(DriverName As String, DriverFileName As String, Output As
String, bRet As Boolean)
Sub OnInstallDriver(DriverName As String, DriverSource As String, Output As
String, bRet As Boolean)
```

The subroutine definitions for customizing upload/download plugins has been modified compared to previous Movicon versions to support new features. Therefore previously created plugins may not work correctly if not adapted to the new features.

The above listed subroutines have the following meanings:

OnLoading
This event is executed for the when the "Upload" command is executed for the plugin selected. This event can be used for initializing the plugin.

OnUnloading
This event is executed at the end when the plugin is unloaded. Therefore it can be executed for the previously used plugin when no other goes into use, or when the "upload" dialog window is closed.

OnConnect
This event is called before initializing each command requiring a device connection. It is actually called in the "Get Device project", "Start Device Project", "Upload Project", "Stop Device Project" and "Erase Device Memory Card". The "ServerName", "UserName", "Password" and "TargetFolder" parameters are filled with values set in the dialog window ("Server", "User Name", "Password"and "Device Folder"). The "bUpload" parameter is set to "True" when the connection takes place because the "Upload Project" button has been pressed to upload project to a device.

OnDisconnect
This event is executed when any operation requesting a connection (OnConnect) has terminated. It is actually called when the "Get Device project", "Start Device Project", "Upload Project", "Stop Device Project" and "Erase Device Memory Card" commands have terminated.

OnCreateDir
This event is executed before a file is uploaded when needing to create a directory in the target. The "DirName" parameter must contain a "\" character at the beginning otherwise the directory may get mistaken for the name of a compact flash from a WinCE device. In cases where the plugin is used for exporting projects to devices other than WinCE, the plugin code must cater for this.

The "OnCreateDir" event can be executed many times with the same "DirName" in cases where there many files in the same folder to upload. The plugin code must be able to handle the possibility that the directory may have already been created.

OnCopyFile
This event is executed for each file that must be copies to the target. The "Source" and "Dest" parameters report the source and destination file path. The "Dest" parameter is composed using the "DirName" which has been initialized in the "OnCreateDir" event. The name of the file to be copied is added to this. The plugin code must use these two parameters to provide a copy of the source file on the target.

OnDeleteFile
This event is executed when needing to delete a file on the target. The "DeleteFile" parameter contains the path and file name to be deleted and is composed using the "DirName" value initialized in the previous "OnCreateDir" event. This doesn't mean that all files to be deleted really exist and therefore the plugin code must take this into account.

OnStartStop
This event is executed when the "Start Device Project", "Stop Device Project" buttons are used in the dialog window for uploading a project. You can use the "bStart" parameter to see of the two Start/Stop command has been executed. The "Project" parameter contains the path and name of the project to be started or stopped. It is composed using the "DirName" initialized by the "OnCreateDir" event which was executed before copying the project file.

OnCreateShortcut
This event is executed when the "Create Device Shortcut" button is used in the dialog window for uploading a project. The "Project" parameter contains the path and name of the project to be created with a link and is composed using the "DirName" initialized by the "OnCreateDir" event before copying the project file.

OnWinVersionInfo
This event is executed when double-clicking a plugin on the list. The plugin must preoccupy and preload the "dwMajorVersion", "dwMinorVersion", "dwBuildNumber", "dwPlatformId" and "szCSDVersion" parameters so that they can be displayed in the appropriate dialog window afterwards.

OnGetFolders
This event is executed for setting the upload path in the dialog window. On the first call the "Path" parameter is empty and the code should fill the "FolderList" parameter with the list of all the folders, or compact flash, read from the device's root (you need to use the "Line Feed" chars as separator for this parameter). At this point a window will display showing the list of all the folders, where the user can choose to which file contents to expand or not. Therefore the "OnGetFolders" event is called every time the user decides to open a folder, and the new path in which the list of folder is to be read is set in the "Path" parameter.

OnCleanMemoryCard
This event is called upon pressing the "Erase Device Memory Card" button. In cases where no path has been specified in the "Upload Device Path" field in the dialog window, this event is preceded by the "OnGetFolders" event call, so that the user can select the folder in which files are to be deleted. The "Path" parameter contains the path to be used for deleting all the files. The "TotalNumber" parameter must be set with the number of files still to be deleted and is needed for managing the progress status bar. The "OnCleanMemory" event is called until the "bContinue" and "bRet" parameters are both at "True". The Output parameter can be used for getting the names of files which have been deleted upon each "OnCleanMemoryCard" event call.

OnGetBootFile
This event is called on the "Get Device Project" command before starting file downloading from the device. the plugin must search for the boot file in the connect device, read it and retrieve the project path. The "BootFile" parameter must be set with the boot file name. The "DevicePath" parameter must be set with the path in which the project is found in the device (read from boot file). The "TotalNumber" parameter must be set with the total number of files to be downloaded, and is used for managing the progress status bar.

OnDownloadFile
This event is called on "Get Device Project" command if the "OnGetBootFile" event has successfully retrieved and read the boot file from the device. The "Path" parameter contains the local path that has been selected in the download window. The "LastFile" parameter must be set with the file just

downloaded. The "OnDownloadFile" event is called until the "bContinue" and "bRet" parameters are both equal to "True".

OnGetDeviceInfo
This event is executed at the end of project file transfers. This code must fill in the "dwPlatform" and "bIsPocketPC" output parameters with the information in the devices platform (i.e. dwPlatform = 0 for x86, dwPlatform = 5 for ARM, bIsPocketPC = true when dealing with a PocketPC) so that the correct driver gets selected for transferring the information.

OnCheckDriver
This event is executed after "OnGetDeviceInfo" and runs a check to see if there is a driver in the device or for driver's version. "DriverName" is the name of the driver (i.e. "ModbusTCP.dll"), "DriverFileName" is the file's complete path which mus be trasferred to the device if bRet = true

OnInstallDriver
This event is executed after "OnCheckDriver" and copies driver from PC desktop to device's "Driver" subfolder.

Note:
The "Output" parameter for customizing the messages which are copied in the upload window trace at the end of each operation.
The "bRet" parameter has the task of notifying the upload manager of operation outcomes. Setting this parameter to "False" will interrupt the execution of the next events.

"Plugin1.updmgr" file example:
This plug-in copies the project to another directory.

```
Sub OnConnect(TargetFolder As String, Output As String, bRet As Boolean)
	TargetFolder = GetFilePath("[Select the target path]", "", CurDir, "Select the target path", 2)
	If TargetFolder = "" Then
		bRet = False
	Else
		TargetFolder = Left(TargetFolder, InStrRev(TargetFolder, "\"))
	End If
	Output = "OnConnect: TargetFolder->" & TargetFolder
End Sub

Sub OnCreateDir(DirName As String, Output As String, bRet As Boolean)
	Dim aFolders() As String
	Dim i As Integer
	DirName = Mid(DirName, 2)
	aFolders = Split(DirName, "\")
	DirName = aFolders(0) & "\"
	For i = 1 To UBound(aFolders)
		DirName = DirName & aFolders(i) & "\"
		On Error Resume Next
		MkDir DirName
		On Error GoTo 0
	Next
	Output = "OnCreateDir: DirName->" & DirName
End Sub

Sub OnCopyFile(Source As String, Dest As String, Output As String, bRet As Boolean)
	Dest = Mid(Dest, 1)
	FileCopy Source, Dest
	Output = "OnCopyFile: Source->" & Source & ", Dest->" & Dest
End Sub

Sub OnDeleteFile(DeleteFile As String, Output As String, bRet As Boolean)
	Kill DeleteFile
	Output = "OnDeleteFile: DeleteFile->" & DeleteFile
End Sub
```

5.6. RAM Use

The Windows CE operating systems allows less resource usage compared to the Win32/64 system. It is for this reason that you should always consider the size of you wish to run on this platform and there resources it needs. One of the big constraints in Windows CE is that this system can only allocate 32 Mb of RAM with WinCE 5.0 and 2GBytes of RAM with WinCE 6.0. **The operating system will terminate any processes going over this RAM use threshold.**
The hardware device must have a sufficient amount of memory to cater for the applied project. The minimum memory required is 32 MB of free RAM, but we strongly advice you to use devices with 64 MB RAM. The project may require more RAM if programmed to use IMDB historicals, VBA logic or other advanced functions.

Each process in WinCE can use a maximum of 32Mb for the 5.0 version and 2Gb for the 6.0 version. Movicon allows this limit to be exceeded when needed by using its Heap Memory management (project general properties). When this management is enabled, Movicon CE will use the Heap memory blocks to exceed the limit set by the WinCE operating system. This is only valid for Windows CE 5.0 or later.

Tests have been run on a device using a X86 processor to give you an idea on the minimum amount of memory a Movicon CE application can occupy. This test was done by creating and running a new empty project without any screens.
The results showed that the following memory was used:

Movicon CE (runtime)	7 MBytes approx.
WinWrap Basic	2,5 MBytes approx.
ADOCE and SSCE	2,3 MBytes (optional)
Total	**11,5 MBytes approx.**
WinCE 5.0 (PRO)	37 MBytes approx.

It is quite clear that the Movicon CE process occupies as little as 11-12 Mbytes compared to the 36 MB of occupied memory used just for the Windows CE 5.0 (PRO vers.) image. The rest of occupied memory will depend on the **project size and the use of the IMDB**. Also consider that additional memory use will be needed when using Basic Script code in your project which can be calculated as follows:

- approximately 700 Kbytes needed for loading the basic script instruction interpreter which is loaded only once for each execution thread
- approximately 300 Kbytes needed for each object or resource containing basic script (screens, symbols, alarms)

To avoid projects using too much memory a Movicon registry key has been inserted to define the minimum amount of memory that must remain free. When this limit is exceeded, the screens will be unloaded from memory even when enabled with "DO NOT Destroy", and the basic script resources will not be loaded in memory. Furthermore the page change will be managed as if set with the '0' value in 'TimeDeferClosingWnd' property. Once reentered from this status every thing will go back to working as before. The key in question is:

General->MaxAvailVirtual (default value 5000000 bytes)

The default value is 5 Mbytes. This means that once the free memory value for the Movicon CE process (32 MByte available for 5.0 and 2GBytes for 6.0.) goes under the 5 Mbytes, Movicon will start behaving as described above.

For further information about the Windows CE memory management click the Microsoft link below:

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dncenet/html/advmemmgmt.asp

Low Memory Condition

The "MaxAvailVirtual" is used for managing the downloading of screens when the Movicon CE application detects insufficient memory.

This condition results when these two cases are evident:

- When memory used by the process nears the "MaxAvailVirtual" value consented by the operating system in use (32 Mbytes for WinCE 5.0, 2Gbytes for WinCE 6.0)
- The the total memory available for programs goes under the level set by the "MaxAvailVirtual" value

When this condition verifies, the opening of a new screen will force the downloading from memory of the first loaded screen. In addition executing a script, it may be denied the same insufficient memory reason.
This condition is also traced with messages in the system log:

- "You are running on low memory condition !", is reported each time Movicon CE acknowledges this condition.
- "Unloading Screen '\flashfisk\...\resources\screen.movscr'. The system is running on a low memory condition", is reported each time a screen is forcefully downloaded from memory.
- "Cannot execute the basic script '<Nom

You need to remember that once Movicon CE enters into the insufficient memory condition, it is difficult to get out as the allocated VM (virtue memory) does not get freed when screens are closed. Forced screen downloading ensures that memory is freed for new screens to use to prevent the Movicon CE application from exceeding the operating system's max. memory which otherwise would force the application to close. In addition, changing pages while in this condition may be rather slow due to the fact that Movicon CE must download a screen from memory before loading the new one.

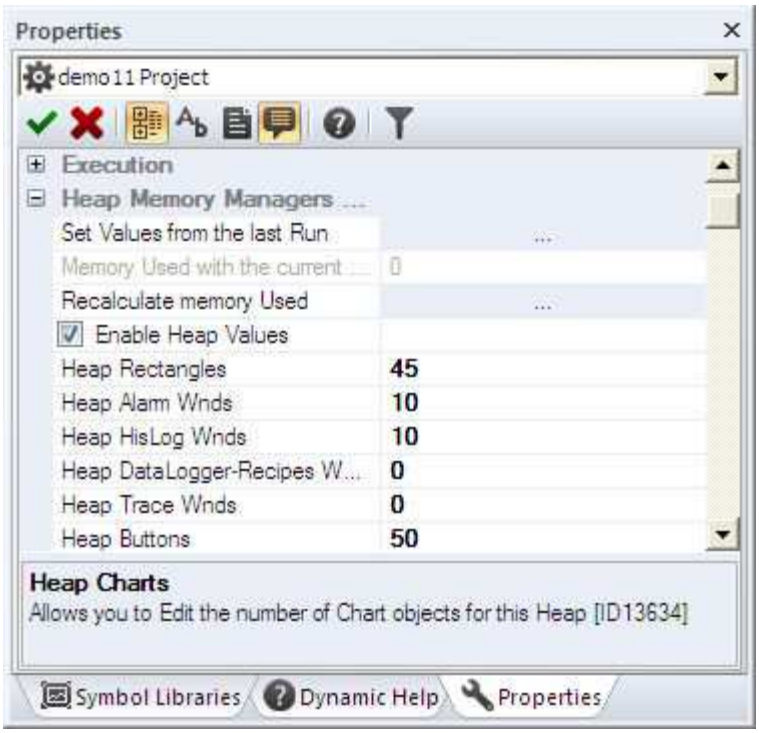
When the system unloads basic script in separate threads while in this condition, it actually frees it from the VM, allowing the application to return back to normal. Unloading scripts in separate threads is not automatically done by the application and must be done manually with the appropriate command.

RAM occupation of one project

Even though it is impossible to know exactly how much memory the project will occupy beforehand. We can, however, presume that a typical project containing approximately 200 variables, 20 screens measuring 640x480 with simple vector graphics, 200 alarms, one communication driver should occupy from 2 to 4 MB of memory (the project is loaded in RAM when run).
This is just purely indicative and is influenced by the graphic types used, the effective number of screens used and by the use of any external files (.BMP or other).

Heap Memory use

Projects that have no choice but to use more than 32 MB of memory, will need to use the "Heap Memory" which has to be activated in the project's properties. Before doing this, be absolutely sure that the project needs to use so much memory and check whether the device is capable of managing all the project's functions. When the application 32 MB memory limit, imposed by Windows CE 5.0, needs to be exceeded, you will need to set the project with the Heap management, which will allow Movicon to fragment the devices memory around the 32 MB limit.



Tips on reducing memory use

These tips will help you reduce memory use on the Movicon CE process side:

- Use bitmaps images instead of jpeg. images which use up more memory because they decompress in memory. Furthermore, when they are used in symbols they remain in the symbols cache and therefore the occupied memory is not freed when the screen gets unloaded
- Activate on those transports you intend to use in the networking settings. This will save you up to 1 Mbytes
- In cases where many data loggers need to be managed (more than 5) it would be best to use the IMDB format instead of the ODBC format, providing that the maximum number of records per data logger is around a hundred.

6. Memory Usage with Historicals

6.1. Memory Use with IMDB

The Movicon CE historicals (Log, Trace, Data Loggers and Recipes) are recored for default in **IMDB** (In Memory DB) mode. When creating projects for WinCE all the historical settings with IMDB are enabled for default in the developing enviroment as the ADOCE components need for recording on SQLCE are not always available in all devices. As an alternative, you can use a relational database, defined as the ODBC (Open Database Connectivity) in the development environment, for recording instead of the IMDB. WinCe does not support ODBC and therefore allODBC connections on the target device are converted automatically in ADOCE connections by Movicon. The SQL Server is the DataBase format set for default.

IMDB Engine

The IMDB engine saves and manages data directly in RAM. The memory tables are unloaded on text files with .dat extensions.
Each historical table has its own file where data is only saved in delayed mode based on a preset time (10 seconds for default) after the table has undergone any modifications. These text files have two functions: they can be used for transferring data to other supported files and are used at the project startup for preloading the tables with their most recent values in the historicals.

The Panel RAM use is also strongly conditioned by the use of the IMDB historicals (InMemoryDB).

There is a "General/InstallComponents" registry key for WinCE that can be set at zero block the ADOCE and SQL Server CE components from being loaded at project start up. This will save memory space if you do not intend to use these components for recording data.

The Movicon CE project's historicals record data using the IMDB engine for default, (except for certain different settings). As an alternative, you can use a relational database for recording instead of the IMDB. If you do not select the IMDB settings in the project, Movicon will use the ODBC (Open DataBase Connectivity) in the Desktop by automatically converting to ADOCE (SQLCE) connections on the panel (WinCE does not have ODBC).
Historical data is recorded by the IMDB engine on output files in the permanent memory (Flash compact) and loaded in RAM at each project startup. The IMDB works in RAM (InMemoryDB) to ensure data access management, analysis, filters and queries in the project.

Default values

When creating a project for WinCE, Movicon will set the following values for default in each of the project's historical recording engines:

1. Use IMDB Manager (property)
2. Shared IMDB Tables
3. Data Max.Age is 180 days (caution, we recommend you reduce this value in panels with little memory capacity)
4. Nr. 4 MB RAM allocation for IMDB manager (MovCE.ini)

- **The default values can or should be changed as required and according to device being use.**

New contraint values for the historicals

- The maximum historical log table age has been set with a new constraints value (in act from build 955) called "MaxDaysAgeHistoric". This value has been set at 7 days for WinCE projects, therefore the alarm, driver and system message tables cannot exceed this maximum age limit in WinCE projects.

Movicon CE will therefore allocate 4MB of RAM for the projects IMDB tables for default. This value can be changed in the MovCE.ini file. You will need to check that:

1. The device has at least 4MB of RAM available.
2. That 4MB of RAM are enough to contain the files required in the project.
3. If the "Shared Tables" setting is disabled Movicon CE will no longer allocate memory and therefore you will need to make sure there is enough RAM to cater for the size of the files set.

It is essential that the recording engines (Historical Log and Data Loggers) are sized according to the amount of data needed: All the IMDB historicals are always stored in RAM which will be consumed during the project run until the preset memory allocated is completely occupied!!

Once this preset limit has been reached (4 MB for default) the IMDB will stop recording any further values until space has been freed from the Historicals' tables. A "IMDB - Internal error: Out of shared memory" message will appear in the system Log.

measure memory consumptions

The IMDB uses the RAM for managing historicals and allocates space to the compact flash for saving .dat and /or xml. files.

The following table shows the correspondence between the types of Movicon project variables and the data type created by the IMDB in its tables:

Data Type	Movicon Data Type	IMDB Bytes Size
Bit, String, Array, Structure	Character	1 byte per character (2 bytes for unicode)
Byte, Sign Byte, Sign Word, Word, Sign Dword, Dword	Numeric	4/8 bytes (32/64 bit value)
Float, Double	Decimal	8 bytes
TimeCol, LocalCol	Date/Time	8 bytes (100 nano-seconds resolute)

Space occupied by a .dat file is about a ratio of 1:3 in respect to memory occupied of the same table. For instance, a table occupying 1 MByte memory space will be about 350 KBytes in size when exported to a text file.
Space occupied by a xml file is about a ratio of 1:3 in repect to memory occupied of there same table. However, you must keep in mind that the xml file is 5 times bigger in size than the dat. file meaning that a noticeable higher amount of memory will be use when saving in xml files.
In principle, an estimated memory use can be calculated using this formula:(Size .dat + Size .xml) * 3

Below is a table showing how much memory will be used according to the number of records to be saved in a .dat or xml file. These value calculations were taken from a Data Logger with 10 mixed columns (Bit, Sign Byte, Byte, Sign Word, Word, Sign Dword, Dword, Float, Double, Strings with 15 chars), in a Vipa XScale PXA255 device .

Records Nr.	Memory size with .dat file	Memory size with .dat and .xml file
100	~ 467 Kbyte	~ 780 Kbyte
1000	~ 865 Kbyte	~ 3940 Mbyte
2000	~ 1292 Mbyte	~ 7448 Mbyte
5000	~ 2737 Mbyte	~ 18038 Mbyte

Note: Using a project with more than one Data Logger IMDB, it's reccomended to set an adequate value as property that define the writing delay time on the storage card, and if possible, different for any Data Logger. This is due to the slow response in writing on a compact flash of certain devices.

It is strongly advised that you use the historicals in ADOCE (SQLCE) when large amounts of data need filing or a big historical file (more than 30 days)

needs to be stored!!

IMDB Setting Options

"Shared Table"	Tables are created in shared memory, which is a chew memory not reserved just for Movicon CE applications but can be shared (default value = true). As WinCE 5.0 is fixed at 32 MB the memory limit that each process can allocate, this option allows the remaining memory to be used in cases where the device's program memory is more than 32 MB.
"Max # Records"	This value can be used to limit the number of records existing on database. Therefore Movicon will see that the maximum number of records in the table is maintained, as from parameter. In any case, data which is older than the maximum age set will be deleted even though the maximum number of records has not been reached. The "0" default value disabled this function. This stops data from being lost when any product updates occur. The recipes are not subject to the max. number of records and age management even though they have these properties.

Some options can be inserted into the "MovCE.ini" file which allow you to customize some aspects of the IMDB.

"IMDBMaxHMemory"	Maximum memory size which can be allocated in the shared area for managing the historicals. The default value is 4 Mbytes for WinCE and 16 Mbytes for Win32/64. Once this limit has been reached the IMDB will no longer record any values until space has been freed in the historical tables. This message will appear in the system log: "IMDB - Internal error: Out of shared memory".
"IMDBMaxLMemory"	Maximum memory size which can be allocated in the local area for managing historicals. This parameter has meaning only when historicals have been set not use the shared area. The default value is 4 Mbytes for WinCE and 64 Mbytes for Win32/64. Once this limit has been reached the IMDB will no longer record any values until space has been freed in the historical tables. This message will appear in the system log: "IMDB - Internal error: Out of local memory".

When using the Movicon Desktop version, local memory is always used for exporting .xml files. When using the WinCE version, the shared memory is always used for exporting .xml files that will, however, be limited by the IMDBMaxHMemory (default 4MB) value.

6.2. Memory Use with SQL Server Mobile

The Relational Database can be used as an alternative to the IMDB for recording data. This relational database has been defined as the ODBC (Open Database Connectivity). Due to the fact that WinCE does not support ODBC all the ODBC connections on the target device are converted to ADOCE connections by Movicon automatically. The SQL Server is the Database format set for default.

- In order to make the ADOCE and SQL Server CE work correctly, you must install their libraries in the device's "Windows" folder first.

ADOCE and SQL Server CE

SQL Server Mobile consents recordings in ""sdf" format (SQL Server).

The "cdb" (Access for CE) format is not supported due to causing to many problems in managing big quantities of data and that only one connection at a time can be managed.

To get the recording on DataBase to work correctly, both the ADOCE and SQL Server libraries need to be installed in the device's "Windows" folder otherwise the system will not be able to manage them and therefore impossible for Movicon to record data in DataBase format. The " Windows" folder is loaded in Ram upon the device's startup and the installation of the ADOCE and SQL Server CE must be implemented in the operating system's image by the product makers. If this is not done so, each time the device is started up the "Windows" folder will be reloaded from the image saved in the Compact Flash and therefore without the ADOCE and SQL Server CE libraries. In this case, if the ADOCE and SQL Server Mobile libraries are in the Movicion CE installation folder, the Movicion CE project will automatically copy these libraries in the Windows folder at startup. These libraries are installed in the Movicon CE installation folder only when MovCE full Setup has been chosen.

To get the recording on DataBase to work correctly, both the ADOCE and SQL Server libraries need to be installed in the device's "Windows" folder.

In order to create a Movicon CE Database file use an empty file to start with. Then in the first project startup phase the file will be copied and the necessary DataBase tables created. This is the reason why the "Empty.sdf" file is uploaded into the Movicon CE installation folder during the installation phase.

Database Files

The Database file is created in the project's "DLOGGERS" folder. Movicon CE creates a Database for each Data Logger or Recipe set up with a "ProjectName_DataLogger/ReicpeName.sdf" name. When Database files cannot be created, an error message will appear in the Output window and in the Status Bar and data will not be recorded in any format.

WARNING! ADOCE does not support Table column names with spaces in. Therefore it would be a good rule of the thumb to avoid inserting spaces in column names. Otherwise an error message will be generated in the Output window and Status Bar when you run the project making it impossible to record data thereafter.

It would be best, especially for Data Loggers, not to set high frequency recording times which may slowdown device performances to the point of causing them to crash. This problem is due to the fact that Compact Flash read/write operations are rather slow in CE devices.

Measured memory consumption

Here are some examples of RAM memory occupating in SQL CE database files:

Type of Device	Data Type	Column Nr.	Bytes per record	10 second for 7 days
SQL Server CE on ARM devices	Bit, Sign Byte, Byte, Sign Word, Word, Sign Dword, Dword, Float, Double, String (15 characters)	10	~ 56	~ 3 MB (Size of the .sdf file) (needs 6 MB availability in the CF, because data elimination starts on the 14th day)
SQL Server CE on x86 devices	Bit, Sign Byte, Byte, Sign Word, Word, Sign Dword, Dword, Float, Double, String (15 characters)	10	~ 148	~ 8,5 MB (Size of the .sdf file) (needs 17 MB availability in the CF, because data elimination starts on the 14th day)

SQL Server Mobile

Movicon CE now uses SQL Server Mobile (SQL Server CE 3.0) instead of SQL Server CE 2.0. Connection to the database is performed by exploiting the "ADOCE 3.1" and the OLE DB Provider "Microsoft.SQLSERVER.MOBILE.OLEDB.3.0" as before.
The files you will need to install in the WinCE device are:

ES:
Dim cnn As Object
Dim rsTemp As Object
Dim sConnectionString As String
Set cnn = CreateObject("ADOCE.Connection.3.1")
Set rsTemp = CreateObject("ADOCE.Recordset.3.1")
sConnectionString = "Provider=Microsoft.SQLSERVER.MOBILE.OLEDB.3.0; Data Source=" & GetDataLoggerRecipePath & DB_NAME_Type1 & ".sdf;"

Those files that need to be installed in the WinCE device are:

- sqlceca30.dll (da registrare)
- sqlcecompact30.dll
- sqlceer30EN.dll (opzionale)
- sqlceoledb30.dll (da registrare)
- sqlceqp30.dll
- sqlceme30.dll
- sqlcese30.dll

Movicon CE is no longer able to guarantee full support to the SQL Server CE 2.0, even with the "General\ADOCEProvider" key set to the "Microsoft.SQLServer.OLEDB.CE.2.0" value.

Memory Use

The new database engine, SQL Server Mobile, offers many advantages in memory use. Memory uses in platforms with Intel-ARM920T-PXA270 (ARMV4I) microprocessors have revealed:

Description	Program Memory
ADOCE + SSCE 3.0	~ 3 MBytes
SSCE Engine 3.0 (General->CompactAndRepairEnabled=True)	~ 750 KBytes
For each Data Logger	~ 1 MBytes

SQL Server Mobile Advantages

The advantages of using SQL Server Mobile are:

1. Databases in SQL Server Mobile (.sdf) format can be analyzed on the desktop as well using the SQL Server 2005 "Microsoft SQL Server Management Studio". This product has the possibility to connect to databases in SSCE 3.0 format by selecting the "SQL Server Mobile"option in the sever connection dialog window. This application requires you to select a SSCE 3.0 database file, after which data can be displayed within by applying "SELECT" queries.

2. Possibility to access the same database with more than one connection at the same time.
3. Possibility to manage databases protected with passwords and data encryption.
4. Database files can hold up to 4 GB.
5. Synchronization is simpler with SQL Server 2005 through data or RDA (Remote Data Access) replications.
6. Databases are automatically compressed during project runs. This is done with the "Auto Shrink" to keep the database more compact where the will delete all empty pages physically when they reach 60% of the file's occupied space.

Converting a SSCE 2.0 database to SSCE 3.0

Movicon CE is not capable of automatically converting SSCE 2.0 databases to the new SSCE 3.0 version.
This conversion is performed using the Microsoft SQL Server Compact Edition "Upgrade.exe" tool. This tool can be installed using the "SqlMobile30DevTools[lang].msi" setup and is installed for default in this path:

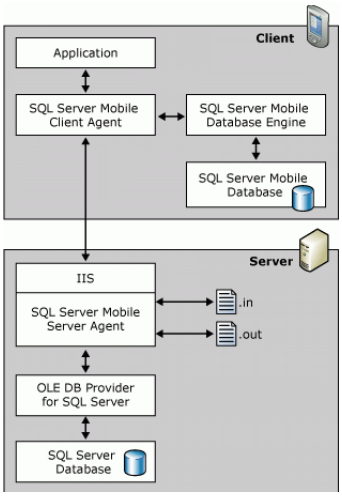
"<drive>:\Program Files\Microsoft Visual Studio 8\SmartDevices\SDK\SQL Server\Mobile\v3.0\[platform]\[processor]\upgrade.exe"

This tool is copied to the device and is executed using the following syntax:

upgrade.exe /s "\dir1\source.sdf" /sp "password1" /d "\dir2\destination.sdf" /dp "password2" /e /q

6.3. RDA (Remote Data Access)

The Microsoft SQL Server 2005 Mobile Edition (SQL Server Mobile) "Remote Data Access" is a way of exchanging data between a remote SQL database and a SQL Server Mobile database.
The "Remote Data Access" (RDA) uses threeMicrosoft SQL Server 2005 Mobile Edition (SQL Server Mobile) components: SQL Server Mobile Database Engine, SQL Server Mobile Client Agent and SQL Server Mobile Server Agent.
In addition, RDA uses Microsoft Internet Information Services (IIS) as means of communication between the SQL Server database in the Server and SQL Server Mobile database in the device. The SQL Server Mobile Database Engine, SQL Server Mobile Client Agent and SQL Server Mobile Server Agent are needed in the RDA management as shown below:



To create a RDA sytem please refer to the Miscosoft (SQL Server Mobile Books Online) documentation.

7. Designing Guidelines

7.1. Variables

The number of variables in a project for Movicon CE should be proportioned to the device's performances. Generally, you need to consider the number of variables the project has **"In use"** and the type of driver being used.
Technically speaking there are no set limits in the programming phase due to the many different factors which determine device performances.
Therefore the programmer will have to decide whether the number of variables to be allocated and managed are adequate according to device's capacity.
It is important how retentive variables are managed as their values, which change constantly, are recorded on the Compact Flash and therefore should be kept to a minimum using only those thought to be indispensable.
Slow Compact Flash access in read/write is quite commonly used in CE devices. Therefore the less it is accessed, the better the device's performances will be. This also goes for the Variable Trace function which should be used only when necessary.

Tips:

1. The number of variables in a project must to proportioned to the panel's capacity. The Movicon CE fixed limit on "Lite" licenses is 1024 bytes and 4096 bytes for standard licenses. However, the designer's own limit must be determined on the panel's potentiality and the communication drivers to be used. It is best not exceed the limit of 512 bytes in use in a 400 Mhx (approx.) panel with PLC communications via serial.

2. Variable retentivity should not be used unless really needed. As values are recorded on the Compact Flash, a good rule of thumb would be to keep a minimum number of necessary variables retentive especially when these change quite frequently. Compact Flash read/write access is quite slow in CE devices and therefore the less the compact flash is accessed the better the device performances will be.

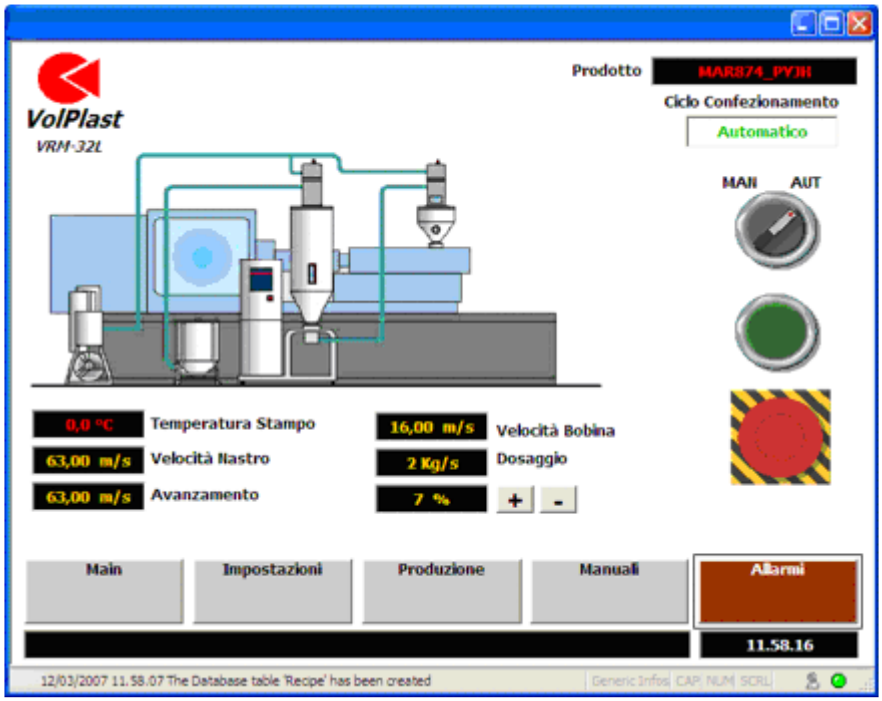
3. 3. The variable Trace function should only be used when really necessary. Each variable value variation traced is recorded on disk with the same concept as that for the retentive variables.

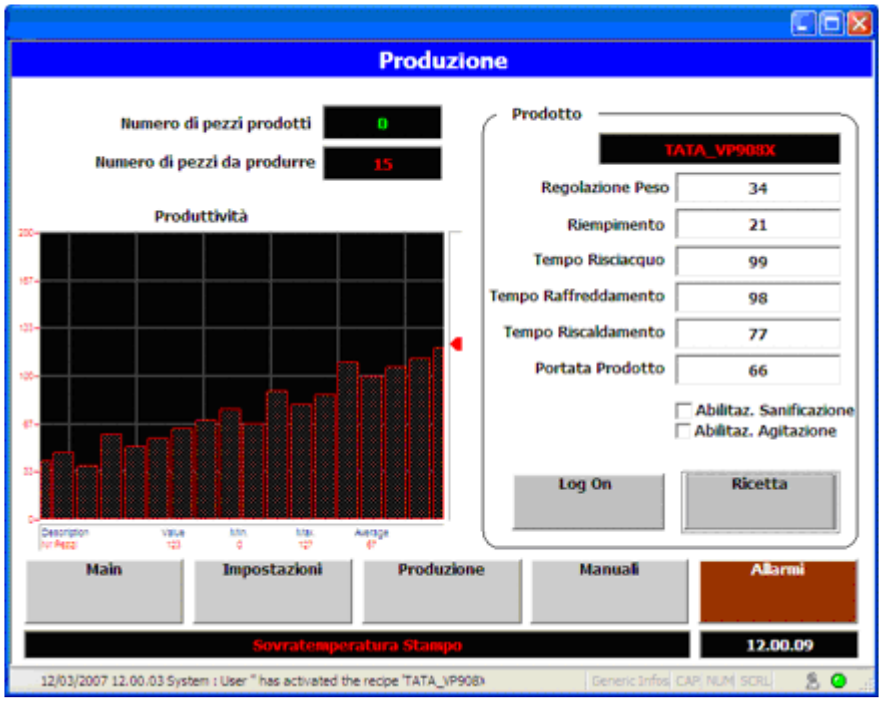
4. The use of dynamic variable via OPC or Networking is onerous for WinCE system resources. Therefore, it would be best to create variables and link them statically using the OPC Client resource or the Tag's Client properties.

5. A good thumb of the rule to stick to would be to map data exchanged in the Word type variables and in the adjoining areas to optimize communications and thus performances. The bit variables can then be addressed to graphic objects or alarms using the name of the word followed by the bit: <name_tag>.<bit>

7.2. The Graphics

Operator panel screen pages should contain simple graphics. Don't get too carried away with the powerful functionalities provided by Movicon without knowing the capabilities of your Touch Panel hardware first. Make sure that the project's platform properties are set to WinCE and go through the Windows CE graphics restrictions indicated in the manual to get a clear picture of the situation beforehand.





Graphic resolution

We suggest you consider the CE device's video screen's resolution. Normally the devices have a limited resolution (640x480 or 800x600) or they can even adopt screens with 1/4 of VGA (320x240). Movicon allows you to adapt the graphics to the video screen's resolution, but if you are drawing with resolutions much bigger than those of the device, the graphics will appear adapted to this resolution but will lose their quality according to the difference between the resolutions (that of the project and that of the device).
Therefore it would be advised to set the video screen resolution or the project screen sizes as near as possible to the devices actual size.

The display area on the target PC, with Windows CE, will certainly be smaller than the desktop PC screen's and for this reason a new command has been introduced called "Remote Device Size

(WinCE)" which is used for sizing the screen window with the characteristics of the target PC to which the project is to be exported. This command only works when connected via Ms ActiveSync. The command can be activated from the Layout menu from any screen.

Before invoking this function you need to setup a connection between the desktop PC and the target PC with ActiveSync.

You can also set the Screen's default settings by reading the device's resolution using the "Get Connected Device Dimensions" command from the project's Platform properties. This can be done every time you create a new Screen to automatically set it with the device sizes.

Below are some advice and suggestions on how to edit projects:

1. Start by setting the sizes of the screens to match the panel screen's resolution (Screen General Properties). Work on how the screen should display the graphics on the panel effectively and see exactly how many objects can be contained on screen within reason. By aiming for better user friendliness you will get better performances. Avoid applying graphic resolutions that are much different from those of the device while designing. For instance, if you are designing screens designated for a 5.7" panel screen in a PC with a 1024x768 resolution you are bound to get poor quality graphics on the panel, with the risk of getting fonts much smaller than foreseen. **Give caution to using a ¼ of VGA (320x240), as screen adaptation from VGA to ¼ of VGA is not possible**.

2. The more graphic objects you put on screen, **the more calculation capacity will be needed to process it.** Try not to use more than 30-40 graphic objects per page unless you are a hundred per cent sure that your hardware device graphics capacity can take it. If in doubt, setup and trial run some pages to test how fast the graphics are on the panel before starting your project.

3. Be careful when using the Movicon symbol libraries. To get their very attractive-to-look-at effect, many of these symbols have been built with an elevated series of vectors. Therefore when you insert symbols from these libraries you must always take into consideration their vector content (even though grouped together) presented on screen. Even though static symbols do not degrade graphic performances, you should always take into account that the page's performances will also depend on its object content (whether static or animated). You can also find and use template objects (ie. Trends) specifically designed for WinCE in the object library.

4. Streamline the use of BMP images to allow only those deemed indispensable. High-resolution BMP images usually take up too much memory space. It would therefore be a good idea to save the 256 color images to reduce occupied memory. However, be warned that not all devices support GIF, JPG formats.

5. To get fast page changes, set the "Do Not Destroy in Run" property = True. By doing this, the displayed screen will remain in memory while other screens are being displayed. More global memory will be occupied but page change will be much faster.

6. Avoid using logic and VBA script in screens. If you do use Script in graphic objects (Power Template), Movicon will have to allocate more memory and process more data, penalizing the project run. If unsure about the potential of your hardware, use the functions provided in the object's standard properties.

7. 8. Avoid using Color gradients, both in the screen background and in the object's background properties. The use of color gradients in backgrounds requires major graphic processing capacity, which will mean a drop in performances.

Supported Images

Movicon CE uses Windows CE APIs for managing all image types. This consents you to manage ".bmp" images with more than 256 colors and other image formats, other than ".bmp", even in devices which do not implement the "imgdecmp.dll" library.

The "Imaging.dll" DLL must be installed in the device to support images. This file is however included in the WinCE 5.0 / 6.0 platform builder.

Change Page

Performances to change one page over to another one may depend on different factors. In general, the time needed to change over the screen is influenced by:

1. The number of objects and components there are on screen
2. The number and sizes of the associated images on screen
3. Compact Flash quality (access time)

The screens are usually unloaded from memory, after a certain presettable time, after which you have to access the relative disk to load them so that they can be displayed. In order to get the best graphic performances it would be best to distribute information adequately on a number of screens as required. When deemed necessary, a screen can always be kept in memory by using the "Not Destroyable" option in its properties. This will make changing screens faster as the screen is always kept in memory even when not displayed but will take up further use of the device's RAM memory resources.
Nevertheless, there are some attributes to insert in the "MovCE.ini" configuration file that allow you to configure certain parameters for managing graphics:

- **UseOffScreenMem**: this parameter allows you to decide whether to use the memory maps for managing screens. The use of the memory maps speeds up screen management (such as zooming operations) at the expense of using higher quantity of memory needed for each screen managed
- **MaxMemoryLoad**: this parameter expresses the percentage of the memory in use that when exceeded will put the screens, in memory but not displayed, into a critical situation causing them to be unloaded and a message will be left in the Output window informing you of this action. The screens will be unloaded even though their "Not Destroyable" option is active
- **ILLogicPriority**: this parameter expresses the priority with which the IL logic is executed. In this case by lowering the IL logic's priority the graphic's execution will be given more priority and viceversa

Font management optimization

To optimize the managing of fonts in WinCE, that in certain cases may be penalizing to system performances, two new keys have been added to be inserted in the "MovCE.ini" Configuration File: "MaxCacheFont" and "DontUseDecorativeFont".

DontUseDecorativeFont

Using certain fonts in devices may greatly penalize device performances. For this reason Movicon CE forces some of the registry keys before creating the font to exclude those parts in the font creation that slowed it down when being used.
Technically speaking, the parts which are forced in WinCE are:

lfOutPrecision = OUT_RASTER_PRECIS
lfClipPrecision = CLIP_STROKE_PRECIS
lfQuality = DRAFT_QUALITY

On the graphical side the created font may seem less attractive but nevertheless is quicker to render. The new **"DontUseDecorativeFont"** value, to be inserted in the "MovCE.ini" Configuration File, restores the original font mode (DontUseDecorativeFont = False).

MaxCacheFont

The option to manage a font cache has been added for further optimization. This cache can be used for storing fonts to be reused whenever needed in order to save Movicon CE the job of creating them over and over again. This additional modification, together with the one mentioned above, should make a great difference in improving performances when many texts are being used on screen. This cache has been set with a maximum number limit to avoid the system from being saturated with fonts when in many different types. This value can be managed by inserting the **"MaxCacheFont"** (default = 0) key into the "MovCE.ini" Configuration File.

All the various project fonts are inserted in the font cache and differ in size and type. Furthermore controls which have been set with the same font may be produced with different sizes if the option for adapting the text to the font size is being used.

For further optimization you can also change the system font. Try to avoid using TrueType fonts (TrueTypes require more processing during the scaling phase). The system font can be changed by using the below

registry key:

"KEY_LOCAL_MACHINE\SYSTEM\GWE\Menu\BarFnt\"

This key contains the system font's facename.

Using images

Using images usually requires allocating memory resources. Therefore it would be in your best interest to check your device's memory capacity and disposition. Big images means more memory use and therefore just use those images which you really need.
Remember that Windows CE also supports a maximum resolution of 256 colors and would be to your advantage to verify the graphical result by setting a resolution of analog colors on the desktop in the programming stage. In addition to this, the Windows CE system colors may be different or less than those of Window 32/64 bit, therefore it would be always best to use the standard colors.

7.3. Alarm Management

MocXCE completely supports the alarm management is completely supported which however carries a few restrictions regarding notifications and statical analysis. Due to the fact that the Report Engine is not supported in Windows CE, the commands relating to creating statical reports on alarms is not available on the CE device.
The alarm notification functions are managed by the Alarm Dispatcher. Only alarm notifications via SMS and E-mail are available in Windows CE as described in the chapter on Restrictions.

Tips:

1. The alarms do not have any particular limits on the way they function and do not require special configurations to optimize them further. However, it is always best to keep the alarm management simple based on the capabilities of your panel.

2. All the alarms are set with the Acknowledge and Reset management and recording on Historical Log for default. Each alarm therefore is managed by 4 events (ON, OFF, ACK, RST): The Acknowledge and/or Reset can be disabled when memory is low, to reduce the number of events managed (and recorded) by the system.

3. If you think you may have too many frequent events, try to avoid recording them in the Historical Log to ensure optimized memory and performances.

4. The Notification of alarms must only be used when deemed necessary and in function with the panel capacity being used. Alarm notification via SMS and Email is supported only.

AlarmDispatcher

In order to send SMS by means of using a GSM modem you need to manually copy the GSM driver configuration file from the Desktop PC to the CE device. This configuration file is found in the Alarm Dispatcher's installation folder with the default name "GsmSMS.stg":

"..\Progea\AlarmDispatcher\GsmSMS.stg"

This file must be copied in the Movicon CE installation folder:

"\Harddisk\MovCE\GsmSMS.stg"

You have to remember that the Alarm Dispatcher version for Windows CE will always search for file called "GsmSMS.stg" only. However, this configuration file can be saved on the Desktop version with a different name. Therefore you need to rename the file with the "GsmSMS.stg" name before transferring it to the CE device.

7.4. Historical Log

The Movicon CE Historical Log is recorded for default in IMDB mode (In Memory DB). When creating projects for WinCE, all the historical log settings are marked with IMDB for default in the development environment to optimize resources by using the XML text format.

Tips on the Historical Log

1. The Historical Log based on IMDB (InMemoryDB) strongly influences the device's RAM. Make sure that the alarm management (see above) is enabled with an adequate Historical Log or use SQLCE if necessary.

2. All the alarms are set with the Acknowledge and Reset management and recording on Historical Log for default. Each alarm therefore is managed by 4 events (ON, OFF, ACK, RST): The Acknowledge and/or Reset can be disabled when memory is low, to reduce the number of events managed (and recorded) by the system.

3. If you think you may have too many frequent events, try to avoid recording them in the Historical Log to ensure optimized memory and performances.

4. It is essential that the Historical Log archive is sized correctly by setting its properties appropriately in the Project properties. As it is impossible to determine how many events can be recorded in a certain period of time, you should always make a wise estimate based on a daily average of recorded events according to the table below. Then, based on the device memory available, set a consistent number of days for data filing (for all three tables).

However you can use the relational database as a data base, which is set with ODBC (Open Database Connectivity) in the development environment. In reality, WinCE does not support ODBC therefore all the ODBC connections are converted by Movicon to ADOCE connections in the target. The SQL Server is set as the DataBase format for default.

In order to for the Historical Log to work in the right way you need to install the ADOCE e SQL Server Mobile, libraries correctly as explained in the paragraph dedicated for this matter

The Database file is created in the project's "LOGS" folder with "ProjectName_HisLog.sdf" as its name. In cases where it is impossible to create the Historical Log in Database format an error message will be generated in the Output window and in the Status Bar, after which the data will be recorded in text format in the project's "LOGS" folder. These files can then be copied and examined on a PC Destktop.

You must remember that more than one connection on one same DataBase cannot be opened at the same time when using ADOCE. This means that when you want to access Historical Log data by means of using the Basic Script functions, you will not be able to create a connection to the Database because the one created by Movicon, to record data, is already active. This restriction is due to the fact that ADOCE is not multi-threading.

7.5. Data Loggers and Recipes

Correct historical management is absolutely necessary for optimizing resources and the RAM on your panel. Therefore read the following tips carefully and decide whether it is more convenient for you to keep the IMDB engine or to use the SQLCE, based on the information reported in the Memory Usages chapter.

Tips

1. Data Loggers based on IMDB (InMemoryDB) may take up substantial device memory space. Make sure that your project's Historical management is handled in parity with the device memory capacity.

2. It is very important that the file archives of each Data Logger are sized correctly by setting the Database properties of each single Data Logger appropriately.

3. To get the right archive sizes, the Data Loggers should record "on time" and not on event or change. This will help you to determine how much memory is occupied for the period of time set. Contrary to this, the programmer should establish the archive time limits by taking care not to risk saturating the device's memory when setting long time periods.

Attention: Names of tables and columns cannot contain spaces or start with a number. Otherwise table creations, using the SQL Server CE or IMDB, will fail generating an error in the Historical Log.

7.6. Trends

The Trends are also supported in Movicon CE. However, you must keep in mind, especially when using Templates from the Symbols Library, that the Trends are rather complex and heavy components for the system to manage. When you use these objects you should always try and follow these tips:

- meet your needs by trying to use the less complex Templates among those available
- set slower sampling times when possible
- do not insert too many pens into one only Template
- set the sampling buffer with the lowest possible value when possible

The "DataBase" button, which is used for linking the Trend to a database which is different from the Project DataLogger, is not available when the Trend Template is exported to Movicon CE. This function is not supported because it needs to exploit DSN connections which are not supported in WinCE.

7.7. Grid

The Grid object is also fully supported on Movicon CE and therefore file contents can be displayed in DataBase format and also in text format (text format files have to be saved as UNICODE and not as ANSI).

The ODBC is not supported in WinCE for the time being and therefore the Database file DSN link cannot be exploited and the Grid's "ODBC DSN" property has to be changed during its configuration phase. This setting will then be different according to whether the Grid has to be linked to a DataLogger/Recipe or to any Database file. In the first case, Movicon will have already opened connections to the file and therefore the Grid should be able to exploit them. In the second case, however, the Grid will have to open a new connection to the DataBase file. As a consequence of this, the "ODBC DSN" property should be set as follows:

- **Connection to DataLogger/Recipe**: the DSN name in the "ODBC DSN" property is replaced with the name of the DataLogger/Recipe
- **Connection to DataBase**: The DSN name in the "ODBC DSN" property is replaced with the DataBase file name to be connected to, without specifying the extension. The DataBase file, in this case, must be found in the project's DLOGGERS folder or one of its sub folders.

The Grid object in Movicon CE connects to a database using the database's ADOCE connection.
Therefore, you must consider that it will not be possible to display databases with the Grid object (Historical Log, DataLogger/Recipe or Trace of variables) if the IMDB is selected as the database.
The IMDB does not use ADOCE connections and therefore not compactible with the Grid object.

7.8. Communication Drivers

The Communication Drivers supported on Movicon CE are generally Serial or Ethernet type. This is due to fact that is difficult to find cards dedicated to communication compatibility with the WinCE environment. For the time being only the Hilscher CIF cards for Profibus DP or CanOpen can be found.
The list of supported drivers is continuously being added to and can be found on the Progea web site.

Many other devices communicate by means of the OPC technology which Movicon CE supports as OPC Client whether a specific DA or a XML DA.

The Communication Drivers are not installed during the Movicon CE installation. This is because the space provided on the CE devices is usually very minimal and not all the driver's dll need downloading. Therefore this job is left to the programmer to install only the ones he/she needs. For further information on installing Communication Drivers please refer to the section on "Installing Movicon CE".

The Communication Driver updates found on the Progea web site do not come in installation files and only the updated dll can be downloaded leaving the programmer to copy it in the appropriate folder on the CE device which will result as:

..\Movicon CE\Drivers

Tips

1. Before designing your project make sure the driver to be used is available for the Window CE version. Not all Movicon drivers can be used technically in Windows CE.

2. The Movicon CE installation on device does not install drivers for reasons of occupying space. The communication drivers used must therefore be installed manually on the device according to the instructions in the manual.

3. Even though Movicon CE allows contemporary communications with a max. of 4 drivers, it would be best just to used one only if unsure that your panel can handle more than one.

4. You can only connect as OPC Client to OPC Servers installed locally on the device with Windows CE. You cannot have a distributed OPC Server network

5. Instead of using Dynamic OPC connections in Tags, it would be better to create links to the OPC Server items in static mode using the OPC Client DA (COM) resource to avoid degrading performances.

6. The Movicon Client configuration is done in the project development phase and you will need to install the same OPC Server installed on the Desktop PC, which will then be used on the target device. Naturally, the two versions of the OPC Server will be specifically for the two WinCE and Win32/64 platforms, but it is essential that they have the same "CLSID" ID code.

7. Above all things, when creating projects for CE, it would be a good habit to create OPC Item groups based on variable usage. For instance, variables that go in use at the same time should be grouped together. This will go towards optimizing performances, which is something that you must always take into account in WinCE. It would also be beneficial to insert groups with "Update Rate" times based on their execution priorities.

8. Before using the driver's advanced functions (VBA or Modem), check carefully which functions are available for the panel you are using.

Driver Performances Benchmark (Example)

Measured SIEMENS S7 TCP Driver performances in VIPA PXA270 420 Mhz Touch Panels:

Number of Words	Refresh Time (sec.)
32	0.39 -0.42
100	0.40 – 0.43
250	0.41 – 0.44
500	0.41 – 0.44
1000	0.42 – 0.46
1500	0.42 – 0.47
2000	0.42 – 0.48

7.9. OPC

Due to the fact that the devices are limited in what they can do, Movicon CE currently supports the OPC Client technology only (OPC Server is available only on Win32/64 platforms). In addition to this, as the DCOM technology is not supported on WinCE, Movicon can connect as Client only to OPC Servers installed locally on the device. It is not possible, however, to have OPC Servers distributed on a network.
The Movicon OPC configuration is done in the project development phase, where you will need to have the same OPC Server installed on your PC Desktop. This will then be used on the Target device. Obviously, the two versions of the OPC Server will be specific to the WinCE and Win32/64, but it is essential that they both have the same identification code "CLSID".

A good rule of the thumb, above all when creating projects for CE, would be to always try and create OPC item groups based on how the variables are used, or rather, to group those which go into use at the same time. This will give way to performance optimization, which is something you cannot ignore when it comes to WinCE. Another thing to do would be to insert a "**Update Rate**" time of the groups according to their execution priorities.

You must be careful when using dynamic OPC connections which are rather heavy for Movicon. Therefore to remedy this problem you should not use them for WinCE, but create static Tags by using the "OPC Client DA (COM)" resource.

7.10. Logics

Movicon CE has VBA (Visual Basic for Application) and IL logic languages (Instruction List). Please take the following into account when you need to use logics in your target device:

Tips

1. The use of the VBA script code may result onerous in panels with processor limits and should be avoided unless absolutely necessary. The Movicon objects provide optimal configuration, command and animation options in their properties.

2. VBA script code should not be used in symbols and screens if possible in some cases, otherwise this may effect screen refresh and change page performances. If needed, you can keep screen in the memory by enabling the screen's "Do not destroy in Run" property, even though this occupies substantial RAM space.

3. Even though Movicon CE ensures VBA language support (therefore the code executed in desktop is the same that can be executed in the target), and with all due reserved for users, Progea cannot guarantee full support to all the many functions, properties, methods and events available on the desktop. Up till now, it has been impossible to carry out tests effectively on the correct execution of each and every instruction on all the different types of devices that exist today.

4. The use of IL Logic in screens may occupy more of the CPU, which in turn will effect the graphical performances. If necessary, (depending on the device type being used) you can

reduce the priority level, assigned to the IL Logic by Movicon, by using the MovCE.ini file's "ILSleep" and IL Priority" parameters.

7.11. Networking

Movicon CE allows you use the networking management both for Client and Server towards any other Movicon project connected in net, whether based on PC or any other WinCE devices.

Tips

1. Only activate Network Settings with transports needed. This will save you up to 1 Mbyte.
2. Always use station IP addresses to identify terminals in network connections between projects.
3. Active network connections have to be in parity with the capacity of the device. Do not exceed 128 TCP connections when unsure about the capacity of your device.
4. The RAS connections (via Modem or VPN) can be used by Movicon if configured in the control panel as network connections and if started up manually. Movicon CE is not capable of starting up RAS connection automatically, therefore this must already be active on the device.

7.12. Web Client

Movicon CE allows Web Client management, where the WinCE panel can carry out the job of Web Server towards Web Clients connected to the device in a TCP-IP network.

Tips:

1. Movicon CE supports the Web Client technology. Web client can only work when the panel has a Web Server. Otherwise you will need to access the Web Client through the Java applet as described in the accompanying manuals.
2. The screen's pages published on Web Client should be managed with a reduced size, furthermore is suggested to disable the option "Fit on Window" on the screen's properties. A good rule should be to never exceed the graphic resolution of the device, publishing a screen on the web. Infact, as more is the size of web screens pages, as more is the memory required on the device for the management. When the memory exceed the limit of 32 MB, WinCE 5.0 end the process Movicon CE.
3. When using the Web Client you must take into account those resources on your panel which carry out Web Server tasks and process data for those clients connected (client connection is limited to a maximum of 2).

8. Running

8.1. Running Projects

Running a project exported to Window CE can be done by creating a link to the MovCE.exe file followed by the path and name of the project. The problem is that each device has customized procedures for executing files upon system startup and there does not exist any standard procedures to follow unless you intervene on the system's registry keys. However, with Pocket-PCs you can copy the link into the "Harddisk\Windows\Startup \" window where the project run is guaranteed at each PC target startup.

At the startup of the project on the target PC, if the software license has not yet been installed, a proposal will be made at the introduction or in the same window to press the "Please, Demo Mode" button to start Movicon CE and the project in Demo modality.

Upon pressing this button the Movicon CE project will then be started by loading the preset screen into the project.
The Exit command must be appropriately predisposed in the project to cater for when the system's menu bar is not made visible.

Automatic Run at Startup

The automatic running of a project upon system startup depends on the type of WinCE platform being used and which hardware has been installed. Therefore the documentation supplied by the device manufacturers should always be consulted. As a rule, you need to specify the device with a command line which provides the MovCE.exe startup with the name of the applied project.
There exist devices which run applications at their startup which are contained in "autoexec.bat" files or specific files setup by their builders.
Unfortunately, there isn't one set of general rules to follow and therefore you will have to use those described in the device's instructions. Once started up, Movicon will run the project specified in the "MovCE.boot" "Boot File" in the Compact Flash's root.
Here below are some examples below:

PocketPC Platforms

In the Pocket PC platforms (eg. Palm PCs such as Compaq iPack, Cassiopeia, etc.) you should only need to copy the link to the "Windows\Startup\" folder to get the project to run at every target PC startup, needless to say this also depends on the product type being used.
For example, when using the Compaq iPAQ you have to proceed in the following way with the "Test_Me.Ink" example project:
Create a link with the Dos EDIT program by creating a "Test_Me.lnk" file and inserting the line:

```
#50"Programs\MovCE\MovCE.exe" "iPaq File Store"\Examples\Test_Me.movprj
```

SSDK Platforms

In SSDK platforms (or operator panels) the operating system does not have a folder for the startup as described above.
Also already mentioned, each device uses customized start-up procedures which are requested from the device builders. for example, if you are using Advantech devices you could use the following procedures:
The operating system executes a series of operations at the startup which are programmed in the windows registry. One of these operations generally runs the "Startup.exe" file, usually placed in the "\Storage Card\Startup\" path which interprets the "Startup.ini" file. The flash memory folder is usually called "Storage Card", but always check the configuration set in the hardware system being used just in case. If possible, you should set the "CF DISK FOLDER NAME" on the "Storage Card" so that the Startup.exe executable does not go into error.

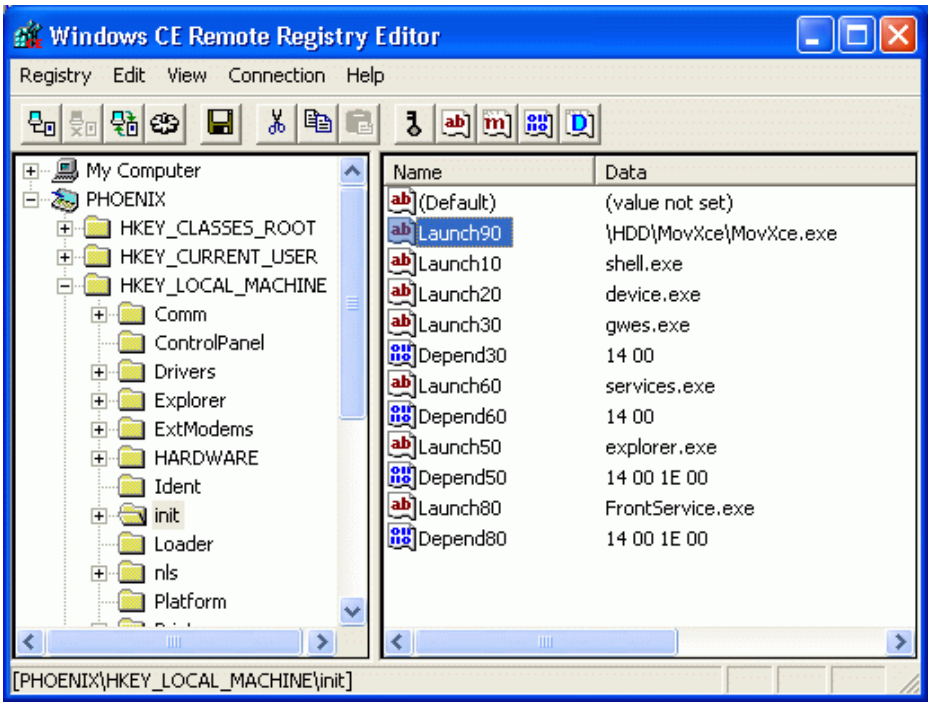
Configuration example used with Advantech TPC 642 panels:
File "Startup.ini:

```
cmd /c \"Storage Card"\Startup\Startup.bat
```

File Startup.bat:
entouch OFF
\"Storage Card"\MovCE\MovCE.exe \IPSM\Me30\test.movprj

Using registries in automatic runs

The method, which allows runs at Start-up using system registry keys, is normally valid for all devices. However, in order to do this you need to be able to access the system registry's configuration, which is usually impossible when not linked up to the right tools, such as the "Windows CE Remote Registry Editor" installed with the "Miscosof eMbedded Visual C++ 4.0" embedded development environment.
The "HKEY_LOCAL_MACHINE\Init" registry key contains the list of executable files to be run at the device's startup. The files are listed under the LauchXX keys, where XX determines the run order. The other key, DependXX, permits the dependency to be specified (eg. Launch50 is not run until Launch40 has been successfully executed).
Normally, but best to check from device to device, you can create a new key under "HKEY_LOCAL_MACHINE\Init", with a progressive number, for example, Launch60 (or a number higher than Launch50), is specified with the name of the executable to be run (note that this must be a .EXE file, such as Startup.exe),in our case this should be "MovCE.exe". If you wish the file to be run only after the other files have been run at startup, you should create a Depend60 and set the value to 32h (hex of 50).
A word of warning: automatic runs may depend on the configuration type of the operating system installed in function with the hardware device. For any further information, please consult the manufacturer's manual of the hardware being used or refer to the Microsoft Platform Builder settings.



Boot File

When Movicon is run, it will search for the "MovCE.boot" file in the root of the Compact Flash at its startup. The file specifying the path and name of the project file to be run should be in this file. The file is in XML format and is structured as follows:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<boot>
<filename>\HDD\PROJECTS\TEST\TESTXCE\testxce.movprj</filename>
</boot>
```

The "MovCE.boot" file is automatically created in the root of the compact flash after the project has been uploaded This happens in automatic only when using the "MS Active Sync" option and by using a different plug-in programmed to manage this function.

After having changed the Device System's registry keys, you should carry out a save of the Registry to avoid losing the changes made at the device's next startup.

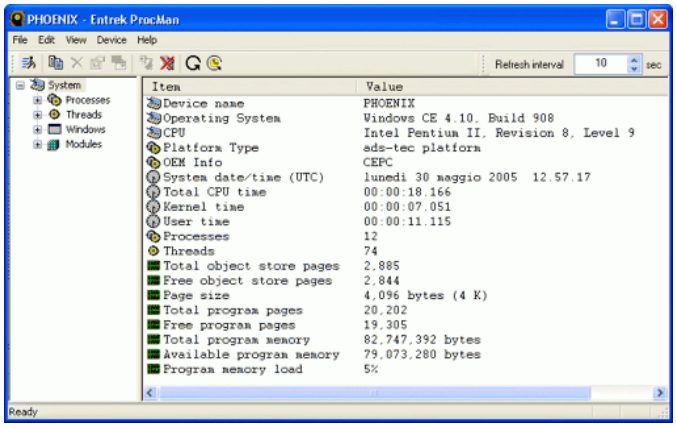
Execute the RunTime module on Windows CE

The Movicon CE installation provides the runtime executive module only for starting and running projects only.
The runtime module is installed in the target, inside the default folder of the applications, or also in other folders if indicated otherwise during the installation. The Movicon CE runtime is identified by the "MovCE.exe" file and by simply double-click it will put it into execution and prepare the next selection of the project to be executed.
As with the Movicon program for desktop, also the runtime module for Windows CE needs a license, the license in this case is software type only. When starting the program up where the software license has not yet been entered, a window will display with the Site Code and in which you can enter the corresponding Site Key to unlock it.
The project can be run in demo mode if the unlock code is not entered and demo reminder window will show at regular timed intervals.
Project run in automatic mode can be realized with a MovCE.exe file link followed with the path indications, project name.

The SSDK devices only have some folders saved on storage memory. The information which is recorded in RAM memory is therefore lost after a next re-start. Therefore we suggest you execute a registry save after the softkey code has been entered otherwise this will also get irretrievably lost.

8.2. Projects Debug

Running a debug on WinCE devices is usually quite limited since the operating system does not provide much information. In addition to the few details on system configuring, as already seen in the paragraph on "Windows CE Operating System", no other information can be got hold of. At this point, if you have a "Platform Manager" development environment installed on your Desktop PC, you can use one of the tools available on the software market, to link up and get more information on the device. 'Entrek ProcMan' is one recommendable tool you could use for displaying system information on active processes, etc.:



Anyway, it is always best to trial run and debug the project well on the Desktop platform first, seeing that it has the most tools available for using. When the Basic functions are being used which are not supported on Movicon CE or are different to those of the Movicon CE, it would be best to use the "RunningOnCE" function to change the code according to the where the project is being run.
Lastly, you can also enable the "Output" window on the Movicon CE, by using the project's "Show the Trace Bar" properties, so that you can consult log messages which are generated by Movicon. In this case it would be best to disable the "Start Full Screen" property to get easier access to the "Output" window.

MOVICON11

Movicon™ is a trademark of Progea, related to the HMI/SCADA platform entirely developed and produced by Progea. © 2012 All Rights reserved.
No part of this document or of the program may be reproduced or transmitted in any form without the express written permission of Progea.
Information in this document is subject to change without notice and is not binding in any way for the company producing it.

Via S.Anna, 88/E
41100 Modena - Italy
Tel. +39 059 451060
Fax +39 059 451061
Email:info@progea.com
Http://www.progea.com

Via XX Settembre, 30
Tecnocity Alto Milanese
20025 Legnano (MI) Italy
Tel. +39 0331 486653
Fax +39 0331 455179
Email: willems@progea.com

Progea Deutschland GmbH
Marie-Curie-Str. 12
D-78048 VS-Villingen
Tel: +49 (0) 7721 / 99 25 992
Fax: +49 (0) 7721 / 99 25 993
info@progea.de

Progea International Ltd
via Penate 16
6850 Mendrisio - Switzerland
tel +41 (91) 9676610
fax +41 (91) 9676611
international@progea.com

Progea USA LLC
2800 East Enterprise Avenue
Appleton, WI 54914
Tel. +1 (888) 305 2999
Fax. +1 (920) 257 4213
info@progea.us